PATENT ABSTRACTS OF JAPAN

(11)Publication number : 09-212090
(43)Date of publication of application : 15.08.1997

(51)Int.Cl. G09C 1/00
H04L 9/14

(21)Application number : 08-039099 (71)Applicant : SONY CORP
(22)Date of filing : 02.02.1996 (72)Inventor : ENARI MASAHIKO

(54) DECODING METHOD AND ELECTRONIC APPARATUS

(57)Abstract:
PROBLEM TO BE SOLVED: To make it possible to complete all of decoding processing in real time on a reception side.
SOLUTION: PID information and TSC information are extracted from the inside of the header of the reception data inputted to a terminal 1 and are supplied to an IDT (inter-data key retriever) 16. This IDT 16 receives such information and searches and reads out data keys out of a DPMEM(dual port memory) 17 by an indirect retrieval method. The plural data keys updated at every prescribed period are written asynchronously in the DPMEM 17. The writing of the DPMEM 17 is prohibited when the writing and reading out of the DPMEM are simultaneously executed with the same address at the same timing. As a resultthe easy memory control of the DPMEM 17 is made possible.

<hr size=2 width="100%" align=center>

CLAIMS

[Claim(s)]
[Claim 1]A write-in step which writes two or more key information in a memory meansand a read-out step which shifts and reads that key information from said memory means to be directed using information in input dataConsist of a decipherment step which decodes said input data based on key information read at this read-out stepand A period of implementation of said write-in stepA decoding method which is a case where a period of implementation of said read-out step lapsand is characterized by forbidding execution of said write-in step when a writing

address and a reading address are in agreement.
[Claim 2]A read-out step which is based on indicative data in input datashiftsand reads that key information from a memory means which has memorized two or more key informationConsist of a decipherment step which decodes said input data based on key information read at this read-out stepand in said read-out step. Key information read from said memory means by referring to a key table with said indicative data is searchedA decoding method choosing fewest key addresses when a key address for reading said key information applicable to said indicative data exists in said two or more key tables.
[Claim 3]By having the following and referring to a table with said indicative data in said read-out stepA decoding method having searched a key address of key information read from said memory meansand searching a key address when a key address referred to with said indicative data does not exist in said table.
A read-out step which is based on indicative data in input datashiftsand reads that key information from a memory means which has memorized two or more key information.
A decipherment step which decodes said input data based on key information read at this read-out step.

[Claim 4]By having the following and referring to a key table with said indicative data in said read-out stepA decoding method characterized by carrying out initial processing in said initial step so that there may be no applicable key addressas a result of searching a key address of key information read from said memory means and searching within an initialization period.
An initial step which performs initial setting.
A write-in step which writes two or more key information in a memory means.
A read-out step which is based on indicative data in input datashiftsand reads that key information from said memory means.
A decipherment step which decodes said input data based on key information read at this read-out step.

[Claim 5]Electronic equipment provided with a decoding means which performs a decoding method according to claim 1 at least.
[Claim 6]Electronic equipment provided with a decoding means which performs a decoding method according to claim 2 at least.
[Claim 7]Electronic equipment provided with a decoding means which performs a decoding method according to claim 3 at least.
[Claim 8]Electronic equipment provided with a decoding means which performs a decoding method according to claim 4 at least.

DETAILED DESCRIPTION

[Detailed Description of the Invention]
[0001]
[Field of the Invention]It is enciphered in this invention.
Thereforein response to the send data by which scramble was carried outit is related with the decoding method and electronic equipment which were decoded.

[0002]
[Description of the Prior Art]In order to keep the information in communication secretthe encryption and the decoding method which acquired the original information are known from the former by enciphering transmit informationand receiving and decoding the enciphered transmit information. Cryptographic algorithmssuch as DES (Data Encryption Standard) which is a standard method in the U.S. as such encryption and a decoding methodare known.
[0003]By the waythere are a variety and various things in a cryptographic algorithmand the method which was more excellent in safety and rapidity is developed. As this examplethe cipher system (MULTI2 method) indicated to the US4982429B Descriptionthe US5103479B DescriptionJPH1-276189Aetc. is known. There are a cipher system registered as ISO9979/0009 also in International Organization for Standardization (ISO) and encryption use mode registered as ISO/IEC10116.
[0004]In the cipher system of the MULTI2 above-mentioned methodinput data size shall be 64 bitsoutput data size is 64 bitsand the work key of 256 bit sizes required in order to perform encryption is generated from the system key of 256 bit sizeand a 64-bit data key. Let the encryption number of stages be a positive integer stage. The outline composition of the encryption algorithm in this MULTI2 method is shown in drawing 17. MULTI2 method generates the 256-bit work key Kw by using 256-bit system key J for the 64-bit data key Ksand calculating a cryptographic algorithmas shown in drawing 17. The operation of this cryptographic algorithm is performed by the cryptographic algorithm execution means C. The 64-bit plaintext which the generated work key Kw was supplied to the cryptographic algorithm execution means Fand was inputted is enciphered. The cryptographic algorithm performed by the cryptographic algorithm execution means C and the cryptographic algorithm execution means F is the same cryptographic algorithm.
[0005]Although such encryption is a fundamental encryption algorithm of MULTI2 methodNowthere is a possibility that a plaintext may be presumedby carrying out the statistical work of the distribution of the frequency where a character or a word appears beforehandand taking matching with the frequency distribution of the character string pattern of the encryption sentence which came to hand. Thenthere is the technique of calculating the exclusive OR of the enciphered 64 bits code block and the 64-bit input data inputted into the nextand creating a cryptogram. The mode

which performs this technique and is enciphered is read with the CBC (Cipher Block Chaining) mode. The cryptographic algorithm in such CBC mode is performed in the above mentioned cryptographic algorithm execution means F.

[0006]Although there is a communication method with which the unit of the data which communicatesfor example like packet communication is decided beforehandwhen the data unit which cannot be divided among the number of bits of 1 block is inputteddata comes to remain in a block cipher system which makes 64 bits 1 block. Thenhe is trying to process the fraction process in OFB (Output Feedback) mode. In this OFB modethe fractional part of data is supplied to the cryptographic algorithm execution means Gand is enciphered using a random number. This random number is generated by the cryptographic algorithm execution means G using the work key Kw. Therebythe cryptogram which makes 64 bits 1 block can be acquired now. The CBC mode and OFB mode are called encryption use mode.

[0007]The outline composition of the decoding algorithm in MULTI2 method is shown in drawing 18. As shown in drawing 18the 256-bit work key Kw is generated by using 256-bit system key J for the 64-bit data key Ksand calculating a cryptographic algorithm. The operation of this cryptographic algorithm is performed by the cryptographic algorithm execution means c. Decoding of the 64-bit cryptogram which the generated work key Kw was supplied to the decipherment algorithm execution means fand was inputted is carried out. The cryptogram enciphered in OFB mode is supplied to the cryptographic algorithm execution means gand decoding is carried out by using the random number generated by the cryptographic algorithm execution means g using the work key Kw. Therebydecoding of the 1-block 64-bit cryptogram can be carried outand a 64-bit plaintext can be obtained. When considered as the CBC modeit is made for the decipherment algorithm execution means f to have a decipherment algorithm in the CBC mode performed.

[0008]Herealthough explanation of encryption use mode is given referring to drawing 19the outline composition of encryption and decoding of the CBC mode is shown in drawing 19 (a)and the outline composition of encryption and decoding of OFB mode is shown in drawing 19 (b). In the CBC modeas shown in drawing 19 (a)i-th plaintext block M(i) is inputted into the exclusive "or" circuit 101and exclusive OR with ciphertext block C (i-1) in front of 1 block delayed with the register (REG) 103 calculates it. The calculated data is enciphered in the cryptographic algorithm execution means 102 by the work key generated based on the data key Ks. This enciphered i-th ciphertext block C(i) is C(i)=EKs (M(i) .EOR.C (i-1)).

It can express. Howeverit means that EKs (m) enciphers m by Ksand it is shown that EOR calculates exclusive OR.

[0009]And it will be transmitted and this ciphertext block C(i) will be received in a receiver. Received ciphertext block C(i) is decoded using the work key generated based on the data key Ks in the decipherment algorithm execution means 111and is supplied to the exclusive **** circuit 113. Ciphertext block C (i-1) in front of 1 block

delayed in the register (REG) 112 is inputted into this exclusive "or" circuit 113and both exclusive **** calculates. At this timethe data key Ks of the transmitting side and a receiver is equalandtherebyi-th plaintext block M(i) is decoded from the exclusive "or" circuit 113. i-th plaintext block M(i) can be expressed as follows.
M(i) =DKs(C(i) .EOR.C(i-1))
Howeverit is shown that DKs (c) carries out decoding of the c by Ks.
[0010]In the time of OFB modei-th plaintext block M(i) is supplied to the exclusive "or" circuit 105. The output of the cryptographic algorithm execution means 104 random-number-ized by the work key generated based on the data key Ks is supplied to this exclusive "or" circuit 105. 1 block of outputs of the cryptographic algorithm execution means 104 are delayed with the register 103and are returned to the cryptographic algorithm execution means 104. Therebyfrom the exclusive "or" circuit 105ciphertext block C(i) enciphered with the random number is outputted.
[0011]And it will be transmitted and this ciphertext block C(i) will be received in a receiver. Received ciphertext block C(i) is supplied to the exclusive **** circuit 114. The output random-number-ized using the work key generated based on the data key Ks in the cryptographic algorithm execution means 115 is supplied to this exclusive **** circuit 114. 1 block of outputs of this cryptographic algorithm execution means 115 are delayed in the register (REG) 112and are returned to the cryptographic algorithm execution means 115. In this casethe random number supplied to the exclusive "or" circuit 114 is equal to the random number supplied to the exclusive "or" circuit 105andtherebyi-th plaintext block M(i) is obtained from the exclusive "or" circuit 114.
[0012]The outline composition of the encryption and the decoding method which has the encryption use mode explained above is shown in drawing 20. In this figurethe transmitting side is equipped with the scrambler 100and the scramble of the input data is carried out by the scrambler 100and it is transmitted. The transmission line of space etc. is spread by this send data by which scramble was carried outand it is received by a receiver. The receiver is equipped with the descrambler 110.
The send data scramble was carried out [send data] by this descrambler 110 is descrambledand it comes to be returned and outputted to the original data.

[0013]Encryptor 102 which is a cryptographic algorithm execution means which enciphers the input data (plaintext) into which the scrambler 100 was inputtedIt comprises an OFB mode encryption section set to the register 103the CBC mode encryption section which consists of the exclusive "or" circuit (EX-OR) 101and Encryptor 104 which are cryptographic algorithm execution means from the exclusive "or" circuit (EX-OR) 105. It has Encryptor 106 which generates a work key from a data key and a system key in the scrambler 100. The generated work key is supplied to Encryptor 102104. By the waysince Encryptor 102Encryptor 104and Encryptor 106 are made into the identical configurationthree Encryptor(s) can be made to serve a

double purpose by one Encryptor. Since operation of the CBC mode encryption section and an OFB mode encryption section is as having mentioned aboveit is omitted here.
[0014]Decryptor 111 which is a decipherment algorithm execution means by which the descrambler 110 carries out decoding of the inputted received data (cryptogram)It comprises an OFB mode decoding part set to the register 112the CBC mode decoding part which consists of the exclusive "or" circuit (EX-OR) 113and Encryptor 115 which are cryptographic algorithm execution means from the exclusive "or" circuit (EX-OR) 114. It has Encryptor 116 which generates a work key from a data key and a system key in the descrambler 110. The generated work key is supplied to Decryptor 111 and Encryptor 115. Since Encryptor 115 and Encryptor 116 are made into the identical configurationtwo Encryptor(s) can be made to serve a double purpose by one Encryptor. Since operation of the CBC mode decoding part and an OFB mode decoding part is as having mentioned aboveit is omitted here.
[0015]
[Problem(s) to be Solved by the Invention]By the waywhen realizing and using advanced encryption and decoding method like MULTI2 method mentioned above by softwarea real-time operation cannot be performed in the computing speed of the present calculating means. that iswhen it appliesfor example to satellite digital television broadcast etc.decoding processing [in / for a picture or a sound / a receiver since / without ******** / it is necessary to come out and carry out real-time reproduction] must be performed by the possible hardware of high-speed processing. Howeverthe hardware which can complete all the decipherment processings in real time in a receiver had the problem of becoming large-sized from performing complicated processing.
[0016]Thenthis invention sets it as the 1st purpose to provide the decoding method which can complete all the decipherment processings for the data by which scramble was carried out with the advanced cipher system using small and cheap hardware in a receiver in real time. This invention sets it as the 2nd purpose to provide small and cheap electronic equipment provided with the decoding means which can complete all the decipherment processings for the data by which scramble was carried out with the advanced cipher system in real time in a receiver.
[0017]
[Means for Solving the Problem]In order to attain the 1st purpose of the abovea decoding method of this inventionA write-in step which writes two or more key information in a memory meansand a read-out step which shifts and reads that key information from said memory means to be directed using information in input dataConsist of a decipherment step which decodes said input data based on key information read at this read-out stepand A period of implementation of said write-in stepIt is a case where a period of implementation of said read-out step lapsand when a writing address and a reading address are in agreementhe is trying to forbid

execution of said write-in step.
[0018]In order to attain the 1st purpose of the aboveother decoding methods of this inventionA read-out step which is based on indicative data in input datashiftsand reads that key information from a memory means which has memorized two or more key informationConsist of a decipherment step which decodes said input data based on key information read at this read-out stepand in said read-out step. Key information read from said memory means by referring to a key table with said indicative data is searchedWhen a key address for reading said key information applicable to said indicative data exists in said two or more key tableshe is trying to choose fewest key addresses.
[0019]In order to attain the 1st purpose of the abovefurther again a decoding method of further others of this inventionA read-out step which is based on indicative data in input datashiftsand reads that key information from a memory means which has memorized two or more key informationConsist of a decipherment step which decodes said input data based on key information read at this read-out stepand in said read-out step. A key address of key information read from said memory means by referring to a table with said indicative data is searchedand when a key address referred to with said indicative data does not exist in said key tableit is made not to search a key address.
[0020]In order to attain the 1st purpose of the abovefurther again a decoding method of further others of this inventionAn initial step which performs initial settingand a write-in step which writes two or more key information in a memory meansA read-out step which is based on indicative data in input datashiftsand reads that key information from said memory meansConsist of a decipherment step which decodes said input data based on key information read at this read-out stepand in said read-out step. A key address of key information read from said memory means by referring to a key table with said indicative data is searchedand as a result of searchingin said initial stepinitial processing is made to be carried out within an initialization period so that there may be no applicable key address.
[0021]He is trying to provide at least electronic equipment of this invention which attains the 2nd purpose of the above with a decoding means which shifts and performs that decoding method to describe above further again.
[0022]Since read-out and writing control of a memory means in which key information required in order to perform decipherment processing is stored can be performed appropriately according to a decoding method of such this inventionit can be considered as a decoding method that memory control can be performed easily. Thereforeit can be considered as a decoding method that all the decipherment processings can be performed in real time in a receiver. In electronic equipment provided with a decoding means which performs such a decoding methodsince composition of a memory control means can be simplifiedelectronic equipment can be provided small and cheaply.

[0023]
[Embodiment of the Invention]The block diagram showing the example of composition of the decipherment device which is an embodiment of the decoding method of this invention is shown in drawing 1. In this figurethe decipherment device A The decoding part B in the CBC modeand the decoding part C in OFB mode. The data key indirect search machine (IDT) 16 provided with a packet ID (PID) tableIt comprises the comparator (COMP) 11 which compares with the writing address and reading address of DPMEM17 the dual port memory (DPMEM) 17 which passes the data key used for encryption of input data to the decoding parts B and C.
[0024]The cryptogram data which is received data is inputted into the terminal 1 of the decipherment device Aand this cryptogram data is inputted into the means for switching 3. When [both] considered as the CBC mode at this timethe means for switching 3 and the means for switching 4 are switched to the terminal a sidedecoding of the inputted cryptogram data is supplied and carried out to the CBC mode decoding part Band it is outputted to it. When considered as OFB modeit is switched to the means for switching 3both the means for switching 4 are switched to the terminal b sideand cryptogram data is inputted into the OFB mode decoding part C side. And decoding of the inputted cryptogram data is carried out in the OFB mode decoding part Cand plaintext data is outputted. In the CBC mode decoding part B and the OFB mode decoding part Calthough decipherment processing of the inputted cryptogram data is carried out and it is considered as plaintext datathe decoding algorithm is the same as the decoding algorithm shown in said drawing 18. A data key required for decipherment processing is supplied from DPMEM17 in this case.
[0025]Let the format of the received data inputted into the terminal 1 be the transport stream (it is hereafter described as TS.) specifiedfor example as ISO/IEC13818. TS is made into 188 bytes of packet structureand he is trying for the pay load of 184 bytes to usually continue after 4 bytes of header. Since parity is added for the error correction to a transmission error16 bytes of straw-man period for parity is addedand it is considered as the stream of these repetitions.
[0026]. [whether the packet is constituted from picture image data by the header of this TSand] The PID information which shows whether it comprises comprising voice data or other dataThe TSC (Transport Scrambling Control) flag etc. which show whether it is TS enciphered are containedand the attribute of these packets is interpreted by analyzing a header in the analyzing parts which are not illustrated. It is made to output TS as it is through a delay partwhen interpreted as the packet which is not enciphered by TSC at this timewithout performing decoding processing. The time delay of this delay part is made equal to the time which decipherment processing takes to the decipherment device A.
[0027]To one PIDtwothe data key Kse (Ks_even) and the data key Kso (Ks_odd)are assigned. Since a data key is updated every [several seconds thru/or] tens of secondsthis needs to rewrite a data key so that it may be enough for updatingAnd it

is because only one of these is usedand updating of the data key which is not used of the data key Kse and the data key Kso is enabled and they can update the data key at the time of decipherment processing. That isit is for performing easily updating control which updates a data key. Although said analyzing parts do not illustrateit prepares for the CBC mode decoding part Bfor exampleand PID/TSC information with a data size of 15 bits is sent to IDT16 from the CBC mode decoding part B.

[0028]In IDT16read address RA (data size is 9 bits) of DPMEM17 is generated combining the address information read from PID with reference to the PID table using the PID information of 13 received bit sizesand the TSC information on 2 bit size. Generated read address RA was supplied to RA terminal of DPMEM17 from AO terminalit was controlled so that a lead enabling (RE) signal would be in an active state at this timeand it has read the data key applicable to read address RA from DPMEM17. The read data key is equal to the data key used at the time of encryption of inputted TSis outputted from terminal RD and supplied to Decryptor 5. Decryptor 5 performs decipherment processing of the cryptogram data which generated the work key and was inputted based on the generated work key from the supplied data key and a different system key for every system. The system key is beforehand passed to the decipherment device A from CPU10.

[0029]When you are trying to be written in DPMEM17 and it writes in the data key which is needed in the decoding parts B and C and which is updated before CPU10 is needed in decipherment processingWrite enable (WE) is made into an active stateand the write data (WD) of a write address (WA) with a data size of 9 bits and a data key with a data size of 8 bits is supplied to DPMEM17. In this caseCPU10 is equipped with ROM or RAM and the data given to the decipherment device A is written in these memories. Since the read data becomes unfixed when writing data key information in DPMEM17 and IDT16 comes to read a data key with the same read address as a write addressthe writing of DPMEM17 has been forbidden at this time.

[0030]For this reasonthe comparator 11AND gate 12OR gates 13 and 14and the inverter 15 are formed. Explanation of this operation will input into the relay terminal and Q terminal of the comparator 11 read address RA outputted from IDT16and the write address WA outputted from CPU10respectively. And when the input data of a relay terminal and the input data of Q terminal are in agreement (P=Q)a high level signal is outputted from the comparator 11. Since high level will be outputted from the inverter 15 if the reversal RE is an active state with a low level at this timehigh level will be outputted from AND gate 12 despite a join office. Since the output of this AND gate 12 is made are inputted into OR gate 13 and high-level [the output of OR gate 13]the reversal WE is inactive and the writing of DPMEM17 will be forbidden.

[0031]Nextalthough the memory space of the register of the decipherment device A seen from CPU10 is shown in drawing 3The area of 64bit x1 where top 6 bits of an address (HADD) with a data size of 9 bits were set to "000100" is assigned to the initial value table (CBCInitial value table) in the CBC mode. At the time of the initial

processing performed by the power up etc.the initial value memorized by this initial value table is supplied to the CBC mode decoding part Band is set in the register 6. The area of 256bit x1 where top 4 bits of an address (HADD) were set to "0010" is assigned to the system key table (SYSTEM_Key table). Although these system keys shall differ for every systemit is the key fixed in one system.

[0032]The area of 13bit x12 where top 4 bits of an address (HADD) were set to "0100" is assigned to the PID table (PID value table). This PID table is information which shows the packet-ized kind of dataand it is referred to as PID which is made into a maximum of 12 kindsand differs for every kind of data in one channel. The information on a PID table is written in a register predetermined in a system key table from CPU10 IDT16 at the time of initial processing.

[0033]The area of 64bit x12 where top 2 bits of an address (HADD) were set to "10" further againIt is assigned to a data key Kse table (Ks_even value table)and the area of 64bit x12 where top 2 bits of an address (HADD) were set to "11" is assigned to the data key Kso table (Ks_odd valuetable). Although the information on this data key Kse table and a data key Kso table is written in DPMEM17it is updated by CPU10 for every prescribed timing for several seconds thru/or several 10 seconds.

[0034]Nextthe block diagram showing an example of the detailed composition of IDT16 is shown in drawing 2. In this figure41-52 are 12 flip-flops (DF0-DF11) which have 13 bit widthrespectivelyand 61-72 are 12 comparators (CP0-CP11) which have 13 bit widthrespectively. 40 is an address decoder (ADEC)and if address information with a data size of 9 bits is inputtedthe output which activates either DF0-DF11 will be taken out only to one in the output of 12. 39 is a priority encoder (PE) and the address information TMP0-TMP3 combined with TSC information is outputted.

[0035]In this IDT16it is developed by the data size of 13 bits and PID information with a data size of 8 bits outputted from CPU10 is inputted into the terminal 30. Address information with a data size of 9 bits outputted from CPU10 is inputted into the terminal 32and it is decoded in ADEC40. This decode output is inputted into DF0-DF11 as a strobe signaland the latch of only any one of DF0-DF11 of it is enabled. At this timePID information with a data size of 13 bits inputted is inputted common to 12 DF0-DF11DF0-one every DF 11 are chosen one by one by the decode output of ADEC40and the PID information currently supplied is latched. Thusa maximum of 12 kinds of PID information can be latched to DF0-DF11and a PID table comes to be constituted by DF0-DF11.

[0036]PID information with a data size of 13 bits is inputted into the terminal 33and it is inputted into it common to B input terminal of the comparators CP0-CP11. This PID information is PID information with a data size [of the PID/TSC information with a data size of 15 bits which was extracted from the header of the packet which is received dataand was supplied to IDT16] of 13 bits in the CBC mode decoding part B. In this casethe PID information from DF0-DF11 is inputted into the generator terminal at the comparators CP0-CP11respectivelyand a coincidence signal is outputted from

the comparator as which the PID information inputted from the terminal 33 and PID information in agreement are inputted into the generator terminal. Thereforeif the PID information in a PID table is inputted from the terminal 33a coincidence signal will be outputted from any one of CP0–the CP11.

[0037]Although this logical table of CP0–CP11 is shown in drawing 41signal is outputted when in agreement [all"1" is not inputted into the generator terminal and battery terminal of CP0–CP11and] (A=B). "0" is outputted when not in agreement. The logical table of PE39 is shown in drawing 5. In the logical table of this PE39when CP2 to "1" signal is outputtedfrom PE39the 4 bits (TMP0–TMP3) data of "0010" is outputted. In this PE39as shown in drawing 5D0 is considered as the input to which priority was given mostand D11 is considered as the input of the lowest priority. By the wayalthough the PID information of the PID table set as DF0–DF11 and conflicting PID information may be inputtedall"0" will be inputted into PE in this case. NK output of PE is set to "1" at this time. When NK output is "1"going to read the key table of the data key mentioned later is forbidden.

[0038]Since the level of the terminal 31 is set to "0" at the time of initial processingssuch as a power upand all of DF0–DF11 are presetDF0–DF11 to "1" is outputted. ThereforeCP0–CP11 to all"0" will be outputted in this caseand going to read the key table of a data key as mentioned above is forbidden. Since a key table is not used when going to read a key table is forbiddena data key can be written in this key table from CPU10. That isinitial setting of a data key can be carried out. The terminal 31 is a power–on–reset terminaland the level returns after predetermined time "1."

[0039]Nextthe operation which searchs the key table of a data key is explained using the PID information and the TSC information in received datareferring to drawing 6. Suppose that "PID–F" was inputted as PID information with a data size of 13 bits from the header of received dataand "10" was inputted as TSC information with a data size of 2 bits at Step S20 shown in drawing 6. This "10" assumes that it is shown that scramble is carried out with the data key Kse. Subsequentlythis PID information is inputted into the generator terminal of CP0–CP11 at Step S21and it is compared with the PID information stored in DF0–DF11. As a resultCP5 to "1" signal is outputted and output TMP of TEMP3–TEMP0 of PE39 [3 . .0] is set to "0101." That isPID–Fis searched. "0101" which is an output of this PE39 is made into the 1st bit from the 4th bit of the address HADD.

[0040]Subsequently2 bits ("10") of TSC information TSC [1..0] are made into the 5th bit of HADDand the 6th bit at Step S22and the address HADD of 6 bit size [8..3] is generated. Thereforethe address HADD [8..3] becomes "100101." And if the address HADD of "100101" generated at Step S23 [8..3] refers a key tabledata key Kse–F of 64 bit width will come to be obtained from Ks_even Table. And the decipherment of a cryptogram inputted based on obtained data key Kse–F is performed with the decipherment device A. Thuswithout preparing the table of 13 bit width corresponding

to PID informationsince it is considered as the indirect search method which searches a data key with the decoding method of this invention indirectlysince what is necessary is just to prepare the key table of 6 bit widththe capacity of a memory can be reduced and it can miniaturize.

[0041]In this wayin a deciphermentsince only one side of Ks_even Table of a key table and Ks_odd Table is usedthe data key of the key table which is not used can be updated. As mentioned aboveCPU10 performs thisbut since the timing of CPU10 and the timing of the decipherment device A are asynchronous and it operatesCPU10 goes the data key which it is asynchronous and is updated to write in. For this reasonin the dual port memory 17 in which a key table is storedthe case where writing and read-out are simultaneously performed by the same address as mentioned above arises. Although the example of timing at this time is shown in drawing 9when the reversal RE of 8 clock width occurs to the timing to illustrateas mentioned abovethe writing of the same address from CPU10 is forbidden in these 8 clock periods.

[0042]By the waythe case where the same PID is stored in the PID table by a certain cause arises. For examplealthough considered as a maximum of 12 kinds per channel of PID informationwhen you do not need 12 kinds of PIDit is not necessary to write 12 kinds of PID in a PID tableand only PID of the kind which carries out necessity will be written in it. Thenthe data of the column of PID which is not written in may turn into the data same by chance as PID. In such a casesince there is a possibility that it may become impossible to read and decode the mistaken data keyin this inventionthis has been prevented as follows.

[0043]As shown in drawing 7when two or more "PID-F" is shown in a PID tablehe is trying for the address HADD of them to give priority to the smaller one. This is because it goes to write in a PID table from the smaller one of the address HADDso the probability which the larger one of the address HADD has mistaken is high. By thisif "PID-F" is inputted as PID information01000101will come to be obtained as the address HADD [8..1]Processing mentioned above and same processing are performed and data key Kse-F of 64 bit width comes to be obtained from Ks_even Table.

[0044]There may be no PID applicable by the PID information from the header of input dataeven if it refers to a PID table. For exampleas shown in drawing 8even if "PID-F" is inputted as PID informationthere is no PID applicable to "PID-F" all over a PID table. In such a casea data key is not readwithout going to read both Ks_even Table of a key tableand Ks_odd Table. In this caseTS which is input data will be outputted as it is.

[0045]Nextalthough the decipherment flow of the decipherment device A shown in drawing 1 is shown in drawing 10if a transport stream (TS) is inputtedit will be judged whether scramble is carried out at Step S10. This judgment is judged by detecting whether the flag in the header which shows that scramble was carried out stands. In this casewhen the flag standsit is judged with the scramble onand it progresses to Step S11and rewriting of a desired flag etc. is performed here. SubsequentlyPID

information is extracted from a header at Step S12and a key table is referred to. The key table in which host interface processing was performed at Step S16 of key table processingand the key table in this case was written in at Step S17 is referred to. Processing of Step S12 is processing shown in said drawing 6.

[0046]Processing of the above step S10 thru/or Step S12 is header control processing. When judged with the scramble off at Step S10TS is outputted as it is. Subsequentlydecipherment processing in the CBC mode is performed at Step S13and it is judged at Step S14 whether the cryptogram by which decipherment processing is carried out is an integral multiple which is 64 bits. When a cryptogram has a fraction and it is judged with Nodecipherment processing in OFB mode is performed about a fractional part at Step S15and the plaintext by which decipherment processing was carried out is outputted. When judged with a 64-bit integral multiple at Step S14the 64-bit plaintext by which decipherment processing was carried out comes to be outputted.

[0047]By the wayexecution of the decipherment flow as shown in drawing 10 is performed by the decoding algorithm shown in said drawing 18. Since the decoding algorithm shown in drawing 18 is as having mentioned aboveit omits herebut the details of composition of performing the decipherment algorithm and cryptographic algorithm in a decoding algorithm are explained with reference to drawing 11 thru/or drawing 15. Drawing 11 shows the composition of cipher processing which performs a cryptographic algorithm. In drawing 11the input data of 64 bit width is divided into top 32 bits data and 32-bit low-ranking dataand is inputted into the first eight steps of codes. Eight steps of this code is considered as the composition by which four steps of operation stages which calculate a function were repeated twice. And in the first rank of the operation stage 20the operation of the function pi 1 is performed to top 32 bits data and the 32-bit low-ranking data which were inputted. Subsequentlyin the 2nd stepthe operation of the function pi 2 is performed to the output of the first rank. In this casethe work key K1 of 32 bit width is inputted into the 2nd stepand the 2nd-step operation is performed using this work key K1.

[0048]In the 3rd stepthe operation of the function pi 3 is performed to the 2nd-step output. In this casethe work key K2 of 32 bit width and K3 are inputted into the 3rd stepand the operation is performed using this work key K2 and K3. Thenin the 4th stepthe operation of the function pi 4 is performed to the 3rd-step output. In this casethe work key K4 of 32 bit width is inputted into the 4th stepand the operation is performed using this work key K4. In the operation stage 21 which performs four steps of operations which furthermore remainthe operation of the function pi 1 is performed to the output from the operation stage 20 in the first rank. Subsequentlyin the 2nd stepthe operation of the function pi 2 is performed to the output of the first rank. In this casework key K5 of 32 bit width is inputted into the 2nd stepand the operation is performed using this work key K5.

[0049]In the 3rd stepthe operation of the function pi 3 is performed to the 2nd-step

output. In this casethe work key K6 of 32 bit width and K7 are inputted into the 3rd stepand the operation is performed using this work key K6 and K7. Thenin the 4th stepthe operation of the function pi 4 is performed to the 3rd-step output. In this casethe work key K8 of 32 bit width is inputted into the 4th stepand the operation is performed using this work key K8. Thusas for eight steps of codesthe data of a total of 64 bit width of top 32 bits and 32 bits of low ranks in which cipher processing was performed is further inputted into 22. As for eight steps of this codethe operation of eight steps of codes mentioned above in 22 and the same operation are performedand the output data by which top 32 bits and a total of 64 bit width of 32 bits of low ranks were randomized is obtained.

[0050]The number of cycles of eight steps of codes is [number of times of not only 2 times but a request] repeatable as illustrated. Output data is randomized highly and can make encryption strength a strong thingso that many this number of times is repeated. the operation of the function currently performed by the operation stage coming out markedly is done to the substitution which transposes the character according to a fixed rule to other charactersand the operation which performs the transposition which replaces an order of a character.

[0051]Nextalthough the composition of the decipherment processing which performs a decipherment algorithm is shown in drawing 12a different point from cipher processing mentioned above is a point of being made to calculate conversely from the output side of the composition of cipher processing. That isin four steps of operation stages 23 of the beginning of eight steps of codesthe work key K8 of 32 bit width was used for the input data in which 64 bit width divided into top 32 bits and 32 bits of low ranks is encipheredand the function pi 4 is calculated. Subsequentlyin the 2nd stepthe work key K7 was used for the output data of the first rankand the function pi 3 is calculated. In the 3rd stepthe work key K6 and K7 were used for the output data of the 2nd stepand the function pi 2 is calculated. In the 3rd stepwork key K5 was used for the output data of the 2nd stepthe function pi 2 was calculatedand the function pi 1 is calculated to the output data of the 3rd step in the 4th step further again.

[0052]Four steps of such operations are similarly performed using the work keys K4-K1 in the operation 24. Also in 25eight steps of codes in which the operation of eight steps of above-mentioned codes is concatenated performand the output data of a total of 64 bit width of top 32 bits and 32 bits of low ranks by which decoding was carried out comes to be obtained. Let repeat frequency of eight steps of codes be a repeat count of eight steps of codes performed in cipher processingand the same number of times.

[0053]Nextit explains in detailraising the operation stage 20 of cipher processing for the details of the operation currently performed in the operation stage to an exampleand referring to drawing 13. In the operation of the function pi 1 of the first rankthe high order bit divided into inputted 32 bits is outputted as it iswithout calculatingand the exclusive OR of a high order bit and a lower bit calculates itand it

is outputted as a lower bit. In the operation of the continuing function pi 2 of the 2nd stepthe work key K1 is added to the data x of 32 bits of low ranksand x+K1 calculates first. Subsequentlywhen x+K1 is set to ythe 1-bit left cyclic shift of the y is carried outy-1 is added to the valueand z is obtained. Nextthe 4-bit left cyclic shift of the z is carried outand the exclusive OR of the value and z is acquired. This result of an operation and top 32-bit exclusive OR calculateand calculated top 32-bit data is outputted. In this case32 bits of low ranks are outputted as it iswithout the inputted data calculating.

[0054]In the operation of the function pi 3 of the 3rd stepthe work key K2 is added to top 32-bit data xand x+K2 calculates first. Subsequentlywhen x+K2 is set to ythe 2-bit left cyclic shift of the y is carried outy+1 is added to the valueand z is obtained. Nextthe 8-bit left cyclic shift of the z is carried outand the exclusive OR a of the value and z is acquired. The work key K3 is added to aand a+K3 calculates. Subsequentlywhen a+K3 is set to bthe 1-bit left cyclic shift of the b is carried out-b is added to the valueand c is obtained. Nextthe logical sum for every bit of a and x and exclusive OR with the value which carried out the 16-bit left cyclic shift of the c are calculated. The exclusive OR of this result of an operation and the data of 32 bits of low ranks is calculatedand the data of 32 bits of calculated low ranks is outputted. Top 32-bit data turns into top 32-bit output data as it iswithout calculating.

[0055]By the operation of the function pi 4 of the 4th stepthe work key K4 is added to the data x of 32 bits of low ranksand x+K4 calculates first further again. Subsequentlywhen x+K4 is set to ythe 2-bit left cyclic shift of the y is carried outand y+1 is added to the value. This result of an operation and top 32-bit exclusive OR calculateand calculated top 32-bit data is outputted. In this casethe data of 32 bits of low ranks is outputted as data of 32 bits of low ranks as it iswithout calculating.

[0056]In the above-mentioned operationby adding the work keys K1-K4 to datasubstitution processing which replaces a character in other characters is performedand the transposition which replaces the position of a character is performed by carrying out the cyclic shift of the data. Thusa plaintext is enciphered by the cryptogram by performing substitution and the algorithm of transposition. When carrying out decodingit can decode to the original plaintext by performing substitution contrary to encryptionand the algorithm of transposition.

[0057]Nextalthough the composition which calculates the function mentioned above is explained still in detailthe example of the function pi 2 shall be raised to drawing 14and shall be explained. In drawing 14in 1st adding machine Add80input data x and the 32-bit work key K1 of 32 bits of low ranks are addedand addition data y is outputted. In the 1st left round shifter 81the 1-bit left cyclic shift of this addition data y is carried outand it is added with the output of the 1st left round shifter 81 in the 2nd adding machine 82. In the 3rd adding machine 84-1 is added to this added resultand the data z calculates. In the 2nd left round shifter 85the 4-bit left cyclic shift of this data z is carried outand it is supplied to the exclusive "or" circuit 86. The

output data of the 2nd left round shifter 85the data zand top 32 bit input data are inputted into this exclusive "or" circuit 86and the exclusive OR of three data calculates. This result of an operation serves as top 32 bit input data inputted into the next step. Low rank 32 bit input data turns into low rank 32 bit input data inputted into the next stepwithout calculating.

[0058]Nextthe composition of the key schedule processing which generates the work key of 256 bit width from the data key of 64 bit width and the system key of 256 bit width is shown in drawing 15. Key schedule processing is considered as the composition to which one-step cascade connection of two steps and one step of operation stage 28 was carried out for four steps of operation stages 26 and 27 as shown in drawing 15. In four steps of operation stages 26 and 27the operation of the function pi 1 is performed in the first rankthe operation of the function pi 2 is performed in the 2nd stepthe operation of the function pi 3 is performed in the 3rd stepand the operation of the function pi 4 is performed in the 4th step.

[0059]Since such an operation algorithm is the same as the algorithm of cipher processing mentioned aboveomit the explanationbut. In key schedule processinginput data is used as a 64-bit data keythe operation of the function pi 1 thru/or the function pi 4 is performed using the 32-bit system keys J1-J8respectivelyand the eight 32-bit work keys K1-K8 are generatedrespectively. Howevernine steps of operations are performed on the wholeand it is different from the algorithm of cipher processing mentioned above in that the function pi 1 is calculated in a final stage.

[0060]The top 32-bit output data after the function pi2 operation of the operation stage 26 is outputted as the work key K132 bits of low rank output data after function pi3 operation is outputted as the work key K2and the top 32-bit output data after function pi4 operation is outputted as the work key K3. 32 bits of low rank output data after the function pi1 operation of the operation stage 27 is outputted as the work key K4The top 32-bit output data after function pi2 operation is outputted as work key K5The top 32-bit output data after function pi3 operation is outputted as the work key K6the top 32-bit output data after function pi4 operation is outputted as the work key K7and 32 bits of low rank output data after the function pi1 operation of the final stage 28 is outputted as the work key K8.

[0061]If the key schedule processing which was mentioned above and which is shown in cipher processing shown in drawing 11 and drawing 13 is referred tothe composition of four steps of operation stagesi.e.an operation algorithmis made equaland it can perform cipher processing or key schedule processing by repeating the operation of four steps of this operation stageand performing it. From thiscipher processing or key schedule processing can be performed by repeating an operation core and performing itif an operation core is considered as the composition which carried out cascade connection of the operation stage of the function pi 1the operation stage of the function pi 2the operation stage of the function pi 3and the operation stage of the function pi 4 as shown in drawing 16 (a). This operation core generates a work key

from the data key and system key which are shown in drawing 20and it is equivalent to Encryptor which is performing cipher processing in the CBC mode and OFB modeand let it be an Encryptor core. In this casethe data keys Ks1–Ks4 and the data keys Ks5–Ks8 are supplied to an Encryptor core by time sharing.

[0062]If drawing 12 is referred tothe composition of four steps of operation stagesi.e.an operation algorithmis made equaland it can perform decipherment processing by repeating the operation of four steps of this operation stageand performing it. From thisdecipherment processing can be performed by repeating a decipherment operation core and performing itif a decipherment operation core is considered as the composition which carried out cascade connection of the operation stage of the function pi 4the operation stage of the function pi 3the operation stage of the function pi 2and the operation stage of the function pi 1 as shown in drawing 16 (b). This decipherment operation core is equivalent to Decryptor which is performing decipherment processing in the CBC mode and OFB mode shown in drawing 20and let it be a Decryptor core. In this casethe data keys Ks8–Ks5 and the data keys Ks4–Ks1 are supplied to a Decryptor core by time sharing. Thusby repeating and performing an Encryptor corethe algorithm of cipher processing and a key schedule can be performedand a decipherment algorithm can be performed by repeating and performing a Decryptor core.

[0063]As mentioned abovealthough the decipherment device which performs the decoding method of this invention was explainedthe electronic equipment of this invention is a tunera TV apparatusetc. which are provided with such a decipherment device at least. Although the above explanation explained the cryptogram of 64 bit blocks as what generates the plaintext of 64 bit blocks using a 64 bits data key and a 256–bit system keythis invention is not limited to these figures and it can be considered as arbitrary numerical values. This invention is not limited to the encryption and a decoding method which repeats the transposition and substitution which were mentioned aboveand can be applied to other encryption and decoding methods.

[0064]

[Effect of the Invention]Since this invention is constituted as mentioned aboveit can be considered as the decoding method that read–out and writing control of a memory means in which the key information for performing decipherment processing is stored can be performed appropriatelyand memory control can be performed easily. Thereforeit can be considered as the decoding method that all the decipherment processings can be performed in real time in a receiver. Since composition of the memory control means in electronic equipment provided with the decoding means which performs such a decoding method can be made smallelectronic equipment can be provided small and cheaply.

DESCRIPTION OF DRAWINGS

[Brief Description of the Drawings]
[Drawing 1]It is a block diagram showing the example of composition of the decipherment device which is an embodiment of the decoding method of this invention.
[Drawing 2]It is a block diagram showing the composition of IDT in the decipherment device shown in drawing 1.
[Drawing 3]It is a chart showing the memory space of the register in the decipherment device seen from CPU shown in drawing 1.
[Drawing 4]It is a chart showing the logical table of the comparators CP0-CP11 of IDT shown in drawing 2.
[Drawing 5]It is a chart showing the logical table of PE of IDT shown in drawing 2.
[Drawing 6]In the decipherment device shown in drawing 1it is a figure for explaining how to search a data key with an indirect search method from the information in the header of input data.
[Drawing 7]It is a figure for explaining operation when PID overlaps all over a PID table.
[Drawing 8]It is a figure for explaining operation in case PID applicable all over a PID table does not exist.
[Drawing 9]It is a figure showing the example of the lead timing of DPMEM in the decipherment device shown in drawing 1.
[Drawing 10]It is a flow chart which shows the decipherment flow of the decipherment device shown in drawing 1.
[Drawing 11]It is a figure showing the composition of cipher processing.
[Drawing 12]It is a figure showing the composition of decipherment processing.
[Drawing 13]It is a figure showing the details of the fundamental function in cipher processing.
[Drawing 14]It is a figure showing the detailed composition for calculating the function pi 2 in a fundamental function.
[Drawing 15]It is a figure showing the composition of key schedule processing.
[Drawing 16]It is a figure showing the composition of an Encryptor core and a Decryptor core.
[Drawing 17]It is a figure showing the algorithm of the conventional encryption.
[Drawing 18]It is a figure showing the algorithm of the conventional decoding.
[Drawing 19]It is a figure showing the composition in the encryption use mode in the CBC mode and OFB mode.
[Drawing 20]It is a figure showing the composition of conventional encryption and decoding method.
[Description of Notations]
1 Received data2 output data34 switching meansand 5 DecryptorSix registersand 79

and 86 An exclusive "or" circuit and 8 Encryptor10 CPU and 11 An AND gateand 13 and 14 A comparator and 12 OR gate15 An inverter16 IDT17 DPMEMand 20-28 The operation stage39 PE40 ADECand 41-52 A flip-flopand 61-72 A comparatorand 8082 and 84 [An adding machineand 81 and 85] [Left round shifter]

(19)日本国特許庁（ＪＰ）　(12) **公開特許公報（Ａ）**　(11)特許出願公開番号

特開平9－212090

(43)公開日　平成９年(1997)８月15日

(51)Int.Cl.[6]		識別記号	庁内整理番号	ＦＩ		技術表示箇所
Ｇ０９Ｃ	1/00	６３０	7259－5Ｊ	Ｇ０９Ｃ	1/00	６３０Ｚ
Ｈ０４Ｌ	9/14			Ｈ０４Ｌ	9/00	６４１

審査請求　未請求　請求項の数８　ＦＤ　（全 27 頁）

(21)出願番号　特願平8－39099

(22)出願日　平成８年(1996)２月２日

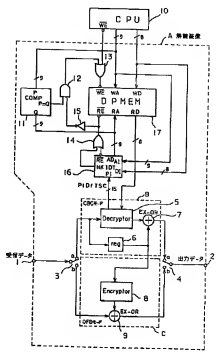
(71)出願人　000002185
ソニー株式会社
東京都品川区北品川６丁目７番35号

(72)発明者　江成　正彦
東京都品川区北品川６丁目７番35号　ソニー株式会社内

(74)代理人　弁理士　脇　篤夫　（外１名）

(54)【発明の名称】　解読化方法および電子機器

(57)【要約】
【課題】受信側においてリアルタイムで全ての解読処理を完了することができるようにする。
【解決手段】端子１に入力された受信データのヘッダ中からＰＩＤ情報とＴＳＣ情報を抽出してＩＤＴ１６に供給する。ＩＤＴ１６はこれらの情報を受けて間接検索方法によりＤＰＭＥＭ１７からデータ鍵をサーチして読み出す。ＤＰＭＥＭ１７には所定期間毎の更新される複数のデータ鍵が非同期で書き込まれており、ＤＰＭＥＭ１７の書き込みと読み出しが同じタイミングでかつ同一アドレスについて行われる時に、その書き込みを禁止する。これにより、ＤＰＭＥＭ１７のメモリ制御を容易に行うことができる。



【特許請求の範囲】
【請求項１】　メモリ手段に複数の鍵情報を書き込む書き込みステップと、
入力データ中の情報により指示されたいずれかの鍵情報を、前記メモリ手段から読み出す読み出しステップと、
該読み出しステップで読み出された鍵情報に基づいて前記入力データを解読する解読ステップとからなり、
前記書き込みステップの実行期間と、前記読み出しステップの実行期間とが重なった場合であって、書き込みアドレスと読み出しアドレスとが一致している場合は、前記書き込みステップの実行を禁止するようにしたことを特徴とする解読化方法。
【請求項２】　複数の鍵情報を記憶しているメモリ手段から、入力データ中の指示データに基づいていずれかの鍵情報を読み出す読み出しステップと、
該読み出しステップで読み出された鍵情報に基づいて前記入力データを解読する解読ステップとからなり、
前記読み出しステップでは、前記指示データにより鍵テーブルを参照することにより、前記メモリ手段から読み出す鍵情報をサーチしており、前記指示データに該当する前記鍵情報を読み出すための鍵アドレスが、前記鍵テーブルに２つ以上存在する場合は、最も少ない鍵アドレスを選択することを特徴とする解読化方法。
【請求項３】　複数の鍵情報を記憶しているメモリ手段から、入力データ中の指示データに基づいていずれかの鍵情報を読み出す読み出しステップと、
該読み出しステップで読み出された鍵情報に基づいて前記入力データを解読する解読ステップとからなり、
前記読み出しステップでは、前記指示データによりテーブルを参照することにより、前記メモリ手段から読み出す鍵情報の鍵アドレスをサーチしており、前記指示データにより参照される鍵アドレスが前記テーブルに存在していない場合は、鍵アドレスのサーチを行わないようにしたことを特徴とする解読化方法。
【請求項４】　初期設定を行う初期ステップと、
メモリ手段に複数の鍵情報を書き込む書き込みステップと、
前記メモリ手段から、入力データ中の指示データに基づいていずれかの鍵情報を読み出す読み出しステップと、該読み出しステップで読み出された鍵情報に基づいて前記入力データを解読する解読ステップとからなり、
前記読み出しステップでは、前記指示データにより鍵テーブルを参照することにより、前記メモリ手段から読み出す鍵情報の鍵アドレスをサーチしており、初期化期間内においては、サーチした結果、該当する鍵アドレスがないように、前記初期ステップにおいて初期処理されることを特徴とする解読化方法。
【請求項５】　請求項１記載の解読化方法を実行する解読手段を少なくとも備えている電子機器。
【請求項６】　請求項２記載の解読化方法を実行する解読手段を少なくとも備えている電子機器。
【請求項７】　請求項３記載の解読化方法を実行する解読手段を少なくとも備えている電子機器。
【請求項８】　請求項４記載の解読化方法を実行する解読手段を少なくとも備えている電子機器。
【発明の詳細な説明】
【０００１】
【発明の属する技術分野】本発明は、暗号化されることによりスクランブルされた送信データを受けて、解読するようにした解読化方法および電子機器に関する。
【０００２】
【従来の技術】通信における情報を秘匿するために、送信情報を暗号化し、暗号化された送信情報を受信して解読することにより、元の情報を得るようにした暗号化・解読化方式が従来から知られている。このような暗号化・解読化方式としては、米国における標準方式であるＤＥＳ（Data Encryption Standard）等の暗号アルゴリズムが知られている。
【０００３】ところで、暗号アルゴリズムには多種・多様なものがあり、より安全性・高速性に優れた方式が開発されている。この一例として、米国特許第４，９８２，４２９号明細書、米国特許第５，１０３，４７９号明細書、および特開平１－２７６１８９号公報等に記載されている暗号方式（MULTI2方式）が知られている。また、国際標準化機構（ＩＳＯ）においてもＩＳＯ９９７９／０００９として登録された暗号化方式や、ＩＳＯ／ＩＥＣ１０１１６として登録された暗号化利用モードがある。
【０００４】上記MULTI2方式の暗号化方式においては、入力データサイズが６４ビット、出力データサイズが６４ビットとされており、２５６ビットサイズのシステム鍵と６４ビットのデータ鍵から、暗号化を行うために必要な２５６ビットサイズのワーク鍵が生成されている。さらに、暗号化段数は正の整数段とされている。このMULTI2方式における暗号化アルゴリズムの概略構成を図１７に示す。MULTI2方式は、図１７に示すように６４ビットのデータ鍵Ｋｓに２５６ビットのシステム鍵Ｊを用いて暗号アルゴリズムの演算を施すことにより２５６ビットのワーク鍵Ｋｗを生成する。この暗号アルゴリズムの演算は暗号アルゴリズム実行手段Ｃにより実行される。生成されたワーク鍵Ｋｗは、暗号アルゴリズム実行手段Ｆに供給されて入力された６４ビットの平文が暗号化される。なお、暗号アルゴリズム実行手段Ｃと暗号アルゴリズム実行手段Ｆとで実行される暗号アルゴリズムは、同一の暗号アルゴリズムである。
【０００５】このような暗号化がMULTI2方式の基本的な暗号化アルゴリズムであるが、これでは予め文字、あるいは単語が出現する頻度の分布を統計処理しておき、入手した暗号化文の文字列パターンの頻度分布とのマッチングを取ることにより、平文が推定されてしまうおそれ

がある。そこで、暗号化された６４ビットの暗号ブロックと、次に入力される６４ビットの入力データとの排他的論理和を演算して暗号文を作成する手法がある。この手法を行って暗号化するモードをＣＢＣ（Cipher Block Chaining)モードとよんでいる。前記した暗号アルゴリズム実行手段Ｆにおいては、このようなＣＢＣモードの暗号アルゴリズムが実行されている。

【０００６】また、例えばパケット通信のように通信を行うデータの単位が予め決められている通信方式があり、６４ビットを１ブロックとするようなブロック暗号化方式では、１ブロックのビット数で割り切れないデータ単位が入力された場合に、データが余ってしまうようになる。そこで、その端数処理をＯＦＢ（Output Feedback）モードで処理するようにしている。このＯＦＢモードでは、データの端数部分が暗号アルゴリズム実行手段Ｇに供給され、乱数を使用して暗号化される。この乱数は、ワーク鍵Ｋｗを用いて暗号アルゴリズム実行手段Ｇにより生成されている。これにより、６４ビットを１ブロックとする暗号文を得ることができるようになる。なお、ＣＢＣモードおよびＯＦＢモードは暗号化利用モードと呼ばれる。

【０００７】また、MULTI2方式における解読化アルゴリズムの概略構成を図１８に示す。図１８に示すように、６４ビットのデータ鍵Ｋｓに２５６ビットのシステム鍵Ｊを用いて暗号アルゴリズムの演算を施すことにより２５６ビットのワーク鍵Ｋｗを生成する。この暗号アルゴリズムの演算は暗号アルゴリズム実行手段ｃにより実行される。生成されたワーク鍵Ｋｗは、解読アルゴリズム実行手段ｆに供給されて入力された６４ビットの暗号文が解読化される。なお、ＯＦＢモードで暗号化されている暗号文は、暗号アルゴリズム実行手段ｇに供給され、ワーク鍵Ｋｗを用いて暗号アルゴリズム実行手段ｇにより生成した乱数を使用することにより解読化される。これにより、１ブロック６４ビットの暗号文を解読化して６４ビットの平文を得ることができる。また、ＣＢＣモードとされている場合は、解読アルゴリズム実行手段ｆがＣＢＣモードの解読アルゴリズムを実行するようにされる。

【０００８】ここで、暗号化利用モードの説明を図１９を参照しながら行うが、図１９（ａ）にＣＢＣモードの暗号化・解読化の概略構成を示し、図１９（ｂ）にＯＦＢモードの暗号化・解読化の概略構成を示している。ＣＢＣモードでは、図１９（ａ）に示すように ｉ 番目の平文ブロックＭ(ｉ) は、排他的論理和回路１０１に入力され、レジスタ（ＲＥＧ）１０３により遅延された１ブロック前の暗号文ブロックＣ(ｉ-1) との排他的論理和が演算される。演算されたデータは暗号アルゴリズム実行手段１０２において、データ鍵Ｋｓに基づいて生成されたワーク鍵により暗号化される。この暗号化された ｉ 番目の暗号文ブロックＣ(ｉ) は、

Ｃ(ｉ) ＝ＥＫｓ（Ｍ(ｉ) ．ＥＯＲ．Ｃ(ｉ-1)）

と表せる。ただし、ＥＫｓ（ｍ）はｍをＫｓで暗号化することを意味しており、ＥＯＲは排他的論理和の演算を行うことを示している。

【０００９】そして、この暗号文ブロックＣ(ｉ) は送信され、受信側において受信されることになる。受信された暗号文ブロックＣ(ｉ) は、解読アルゴリズム実行手段１１１においてデータ鍵Ｋｓに基づいて生成されたワーク鍵を用いて解読され、排他的論和回路１１３に供給される。この排他的論理和回路１１３にはレジスタ（ＲＥＧ）１１２において遅延された、１ブロック前の暗号文ブロックＣ(ｉ-1) が入力されて、両者の排他的論和が演算される。この時、送信側と受信側のデータ鍵Ｋｓは等しく、これにより、排他的論理和回路１１３から ｉ 番目の平文ブロックＭ(ｉ) が解読される。ｉ番目の平文ブロックＭ(ｉ) は次のように表せる。

Ｍ(ｉ) ＝ＤＫｓ（Ｃ(ｉ) ．ＥＯＲ．Ｃ(ｉ-1)）

ただし、ＤＫｓ（ｃ）はＫｓでｃを解読化することを示している。

【００１０】また、ＯＦＢモード時では、ｉ番目の平文ブロックＭ(ｉ) は排他的論理和回路１０５に供給される。この排他的論理和回路１０５には、データ鍵Ｋｓに基づいて生成されたワーク鍵により乱数化された暗号アルゴリズム実行手段１０４の出力が供給されている。なお、暗号アルゴリズム実行手段１０４の出力は、レジスタ１０３により１ブロック遅延されて暗号アルゴリズム実行手段１０４に戻されている。これにより、排他的論理和回路１０５からは乱数により暗号化された暗号文ブロックＣ(ｉ) が出力される。

【００１１】そして、この暗号文ブロックＣ(ｉ) は送信され、受信側において受信されることになる。受信された暗号文ブロックＣ(ｉ) は、排他的論和回路１１４に供給される。この排他的論和回路１１４には、暗号アルゴリズム実行手段１１５においてデータ鍵Ｋｓに基づいて生成されたワーク鍵を用いて乱数化された出力が供給されている。この暗号アルゴリズム実行手段１１５の出力は、レジスタ（ＲＥＧ）１１２において１ブロック遅延されて暗号アルゴリズム実行手段１１５に戻されている。この場合、排他的論理和回路１１４に供給される乱数は、排他的論理和回路１０５に供給される乱数と等しく、これにより、排他的論理和回路１１４から ｉ 番目の平文ブロックＭ(ｉ) が得られる。

【００１２】以上説明した暗号化利用モードを有する暗号化・解読化方式の概略構成を図２０に示す。この図において、送信側にはスクランブラ１００が備えられており、スクランブラ１００により入力データがスクランブルされて送信されている。このスクランブルされた送信データは、空間等の伝送路を伝播されて受信側で受信される。受信側には、デスクランブラ１１０が備えられており、このデスクランブラ１１０によりスクランブルさ

れた送信データがデスクランブルされて、元のデータに戻され出力されるようになる。

【0013】スクランブラ100は、入力された入力データ（平文）を暗号化する暗号アルゴリズム実行手段であるEncryptor 102と、レジスタ103と、排他的論理和回路（EX－OR）101からなるCBCモード暗号化部と、暗号アルゴリズム実行手段であるEncryptor 104と、排他的論理和回路（EX－OR）105からなるOFBモード暗号化部から構成されている。なお、データ鍵とシステム鍵からワーク鍵を生成するEncryptor 106もスクランブラ100内に備えられている。生成されたワーク鍵はEncryptor 102，104に供給される。ところで、Encryptor 102、Encryptor 104、Encryptor 106は同一構成とされているので、1つのEncryptor により3つのEncryptor を兼用することができる。CBCモード暗号化部およびOFBモード暗号化部の動作は前述したとおりであるので、ここでは省略する。

【0014】また、デスクランブラ110は、入力された受信データ（暗号文）を解読化する解読アルゴリズム実行手段であるDecryptor 111と、レジスタ112と、排他的論理和回路（EX－OR）113からなるCBCモード解読化部と、暗号アルゴリズム実行手段であるEncryptor 115と、排他的論理和回路（EX－OR）114からなるOFBモード解読化部から構成されている。なお、データ鍵とシステム鍵からワーク鍵を生成するEncryptor 116もデスクランブラ110内に備えられている。生成されたワーク鍵はDecryptor 111と、Encryptor 115に供給される。なお、Encryptor 115、Encryptor 116は同一構成とされているので、1つのEncryptor により2つのEncryptor を兼用することができる。また、CBCモード解読化部およびOFBモード解読化部の動作は前述したとおりであるので、ここでは省略する。

【0015】

【発明が解決しようとする課題】ところで、上述したMULTI2方式のような高度な暗号化・解読化方式を、ソフトウェアで実現して使用する場合には、現在の演算手段の演算速度ではリアルタイム処理を行うことができない。すなわち、例えば衛星ディジタルテレビジョン放送等に適用した場合は、画像や音声を途切らせないでリアルタイム再生する必要があることから、受信側における解読化処理は高速な処理の可能なハードウェアで行わなければならないことになる。しかしながら、受信側においてリアルタイムで全ての解読処理を完了することができるハードウェアは、複雑な処理を行うことから大型なものになるという問題点があった。

【0016】そこで、本発明は高度な暗号化方式でスクランブルされたデータを、受信側においてリアルタイムで全ての解読処理を小型かつ安価なハードウェアを用いて完了することができる解読化方法を提供することを第1の目的としている。また、本発明は高度な暗号化方式でスクランブルされたデータを、受信側においてリアルタイムで全ての解読処理を完了することができる解読化手段を備える小型かつ安価な電子機器を提供することを第2の目的としている。

【0017】

【課題を解決するための手段】上記第1の目的を達成するために、本発明の解読化方法は、メモリ手段に複数の鍵情報を書き込む書き込みステップと、入力データ中の情報により指示されたいずれかの鍵情報を、前記メモリ手段から読み出す読み出しステップと、該読み出しステップで読み出された鍵情報に基づいて前記入力データを解読する解読ステップとからなり、前記書き込みステップの実行期間と、前記読み出しステップの実行期間とが重なった場合であって、書き込みアドレスと読み出しアドレスとが一致している場合は、前記書き込みステップの実行を禁止するようにしている。

【0018】また、上記第1の目的を達成するために、本発明の他の解読化方法は、複数の鍵情報を記憶しているメモリ手段から、入力データ中の指示データに基づいていずれかの鍵情報を読み出す読み出しステップと、該読み出しステップで読み出された鍵情報に基づいて前記入力データを解読する解読ステップとからなり、前記読み出しステップでは、前記指示データにより鍵テーブルを参照することにより、前記メモリ手段から読み出す鍵情報をサーチしており、前記指示データに該当する前記鍵情報を読み出すための鍵アドレスが、前記鍵テーブルに2つ以上存在する場合は、最も少ない鍵アドレスを選択するようにしている。

【0019】さらにまた、上記第1の目的を達成するために、本発明のさらに他の解読化方法は、複数の鍵情報を記憶しているメモリ手段から、入力データ中の指示データに基づいていずれかの鍵情報を読み出す読み出しステップと、該読み出しステップで読み出された鍵情報に基づいて前記入力データを解読する解読ステップとからなり、前記読み出しステップでは、前記指示データによりテーブルを参照することにより、前記メモリ手段から読み出す鍵情報の鍵アドレスをサーチしており、前記指示データにより参照される鍵アドレスが前記鍵テーブルに存在していない場合は、鍵アドレスのサーチを行わないようにしている。

【0020】さらにまた、上記第1の目的を達成するために、本発明のさらに他の解読化方法は、初期設定を行う初期ステップと、メモリ手段に複数の鍵情報を書き込む書き込みステップと、前記メモリ手段から、入力データ中の指示データに基づいていずれかの鍵情報を読み出す読み出しステップと、該読み出しステップで読み出された鍵情報に基づいて前記入力データを解読する解読ステップとからなり、前記読み出しステップでは、前記指

示データにより鍵テーブルを参照することにより、前記メモリ手段から読み出す鍵情報の鍵アドレスをサーチしており、初期化期間内においては、サーチした結果、該当する鍵アドレスがないように、前記初期ステップにおいて初期処理されるようにしている。

【００２１】さらにまた、上記第２の目的を達成する本発明の電子機器は、上記したいずれかの解読化方法を実行する解読手段を少なくとも備えるようにしている。

【００２２】このような本発明の解読化方法によれば、解読処理を行うために必要な鍵情報が格納されているメモリ手段の読み出しおよび書き込み制御を適切に行うことができるため、メモリ制御を容易に行うことのできる解読化方法とすることができる。したがって、受信側においてリアルタイムで全ての解読処理を行うことのできる解読化方法とすることができる。また、このような解読化方法を実行する解読手段を備える電子機器においては、メモリ制御手段の構成を簡単化することができるので、電子機器を小型かつ安価に提供することができるようになる。

【００２３】

【発明の実施の形態】本発明の解読化方法の実施の形態である解読装置の構成例を示すブロック図を図１に示す。この図において、解読装置ＡはＣＢＣモードの解読化部Ｂと、ＯＦＢモードの解読化部Ｃと、パケットＩＤ（ＰＩＤ）テーブルを備えるデータ鍵間接検索器（ＩＤＴ）１６と、入力データの暗号化に使用されたデータ鍵を解読化部Ｂ，Ｃに渡すデュアルポートメモリ（ＤＰＭＥＭ）１７と、ＤＰＭＥＭ１７の書き込みアドレスと読み出しアドレスとを比較する比較器（ＣＯＭＰ）１１から構成されている。

【００２４】解読装置Ａの端子１には受信データである暗号文データが入力され、この暗号文データは切換手段３に入力される。この時、ＣＢＣモードとされた場合は、切換手段３、および切換手段４が共に端子ａ側に切り換えられ、入力された暗号文データはＣＢＣモード解読化部Ｂに供給されて解読化されて出力される。また、ＯＦＢモードとされた時は、切換手段３、および切換手段４は共に端子ｂ側に切り換えられて、暗号文データはＯＦＢモード解読化部Ｃ側に入力される。そして、入力された暗号文データは、ＯＦＢモード解読化部Ｃにおいて解読化されて平文データが出力される。ＣＢＣモード解読化部Ｂ、およびＯＦＢモード解読化部Ｃにおいて、入力された暗号文データは解読処理されて平文データとされるが、その解読化アルゴリズムは、前記図１８に示す解読化アルゴリズムと同様である。なお、この場合解読処理に必要なデータ鍵はＤＰＭＥＭ１７から供給される。

【００２５】また、端子１に入力される受信データのフォーマットは、例えばＩＳＯ／ＩＥＣ１３８１８として規定されているトランスポートストリーム（以下、ＴＳと記す。）とされる。ＴＳは１８８バイトのパケット構造とされており、通常４バイトのヘッダのあとに１８４バイトのペイロードが続くようにされている。さらに、伝送エラーに対するエラー訂正のためにパリティーを付加することから、１６バイトのパリティ用のダミー期間が付加されて、これらの繰返しのストリームとされる。

【００２６】このＴＳのヘッダにはパケットが映像データで構成されているのか、音声データで構成されているのか、あるいは他のデータで構成されているのかを示すＰＩＤ情報や、暗号化されているＴＳなのか否かを示すＴＳＣ（Transport Scrambling Control）フラグ等が含まれており、図示しない解析部においてヘッダを解析することにより、これらのパケットの属性が解釈されている。この時、ＴＳＣにより暗号化されていないパケットと解釈された場合は、解読化処理を施すことなくＴＳを遅延部を通してそのまま出力するようにする。この遅延部の遅延時間は、解読装置Ａが解読処理に要する時間と等しくされる。

【００２７】また、１つのＰＩＤに対しては、データ鍵Ｋｓｅ（Ks_even）とデータ鍵Ｋｓｏ（Ks_odd）との２つが割り当てられている。これは、データ鍵が数秒ないし数十秒毎に更新されるため、更新に間に合うようにデータ鍵の書き替えを行う必要があること、および、解読処理時にはデータ鍵Ｋｓｅとデータ鍵Ｋｓｏはその一方しか使用されず、使用されていないデータ鍵を更新可能として、そのデータ鍵を更新することができるからである。すなわち、データ鍵を更新する更新制御を容易に行うためである。なお、前記解析部は図示しないが、例えばＣＢＣモード解読化部Ｂに備えられており、ＣＢＣモード解読化部Ｂから１５ビットのデータサイズのＰＩＤ／ＴＳＣ情報がＩＤＴ１６に送られている。

【００２８】ＩＤＴ１６においては、受けた１３ビットサイズのＰＩＤ情報を用いてＰＩＤテーブルを参照し、ＰＩＤから読出されたアドレス情報と、２ビットサイズのＴＳＣ情報とを組み合わせてＤＰＭＥＭ１７の読出アドレスＲＡ（データサイズは９ビット）を生成している。生成された読出アドレスＲＡはＡＯ端子からＤＰＭＥＭ１７のＲＡ端子に供給され、この時にリードイネーブル（ＲＥ）信号がアクティブ状態になるよう制御して、読出アドレスＲＡに該当するデータ鍵をＤＰＭＥＭ１７から読み出している。読み出されたデータ鍵は、入力されたＴＳの暗号化時に使用されたデータ鍵と等しく、端子ＲＤから出力されてDecryptor ５に供給される。Decryptor ５は、供給されたデータ鍵と、システム毎に異なるシステム鍵からワーク鍵を生成して、生成されたワーク鍵に基づいて入力された暗号文データの解読処理を行う。なお、システム鍵は予めＣＰＵ１０から解読装置Ａに渡されている。

【００２９】また、ＣＰＵ１０は解読化部Ｂ，Ｃで必要とされる更新されるデータ鍵を、解読処理において必要

になる前にＤＰＭＥＭ１７に書き込むようにされており、書き込む場合には、ライトイネーブル（ＷＥ）をアクティブ状態とすると共に、９ビットのデータサイズの書込アドレス（ＷＡ）と、８ビットのデータサイズのデータ鍵の書込データ（ＷＤ）をＤＰＭＥＭ１７に供給している。この場合、ＣＰＵ１０にはＲＯＭやＲＡＭが備えられており、これらのメモリに解読装置Ａに与えるデータが書き込まれている。なお、ＤＰＭＥＭ１７にデータ鍵情報を書き込む時に、書込アドレスと同一の読出アドレスでＩＤＴ１６がデータ鍵を読み出すようになった場合は、読み出されたデータが不定となることから、この時はＤＰＭＥＭ１７への書き込みを禁止している。

【００３０】このために、比較器１１とアンドゲート１２、オアゲート１３，１４、インバータ１５が設けられている。この動作を説明すると、ＩＤＴ１６から出力される読出アドレスＲＡと、ＣＰＵ１０から出力される書込アドレスＷＡとが比較器１１のＰ端子およびＱ端子にそれぞれ入力される。そして、Ｐ端子の入力データとＱ端子の入力データとが一致（Ｐ＝Ｑ）した時に、比較器１１からハイレベル信号が出力される。この時、反転ＲＥがローレベルでアクティブ状態になっていると、インバータ１５からハイレベルが出力されるので、結局のところアンドゲート１２からハイレベルが出力されることになる。このアンドゲート１２の出力はオアゲート１３に入力されて、オアゲート１３の出力がハイレベルとされるので、反転ＷＥは非アクティブ状態となり、ＤＰＭＥＭ１７への書き込みが禁止されることになる。

【００３１】次に、ＣＰＵ１０から見た解読装置Ａのレジスタのメモリ空間を図３に示すが、９ビットのデータサイズのアドレス（ＨＡＤＤ）の上位６ビットが“０００１００”とされた６４bit ×１のエリアが、ＣＢＣモードの初期値テーブル（CBC Initial value table）に割り当てられている。この初期値テーブルに記憶される初期値は、電源投入時等に実行される初期処理時に、ＣＢＣモード解読化部Ｂに供給されてレジスタ６にセットされる。また、アドレス（ＨＡＤＤ）の上位４ビットが“０００１”とされた２５６bit ×１のエリアが、システム鍵テーブル（SYSTEM_Key table）に割り当てられている。このシステム鍵はシステム毎に異なるものとされるが、１つのシステムでは固定された鍵である。

【００３２】さらに、アドレス（ＨＡＤＤ）の上位４ビットが“０１００”とされた１３bit ×１２のエリアが、ＰＩＤテーブル（PID value table）に割り当てられている。このＰＩＤテーブルはパケット化されたデータ種類を示す情報であり、１チャンネルでは最大１２種類とされデータ種類毎に異なるＰＩＤとされる。なお、システム鍵テーブルは所定のレジスタに、ＰＩＤテーブルの情報はＩＤＴ１６に、初期処理時にＣＰＵ１０から書き込まれる。

【００３３】さらにまた、アドレス（ＨＡＤＤ）の上位２ビットが“１０”とされた６４bit ×１２のエリアが、データ鍵Ｋｓｅテーブル（Ks_even value table）に割り当てられ、アドレス（ＨＡＤＤ）の上位２ビットが“１１”とされた６４bit ×１２のエリアが、データ鍵Ｋｓｏテーブル（Ks_odd value table）に割り当てられている。このデータ鍵Ｋｓｅテーブル、およびデータ鍵Ｋｓｏテーブルの情報はＤＰＭＥＭ１７に書き込まれるが、数秒ないし数１０秒の所定タイミング毎に、ＣＰＵ１０により更新されている。

【００３４】次に、ＩＤＴ１６の詳細構成の一例を示すブロック図を図２に示す。この図において、４１～５２はそれぞれ１３ビット幅を有する１２個のフリップフロップ（ＤＦ０～ＤＦ１１）であり、６１～７２はそれぞれ１３ビット幅を有する１２個の比較器（ＣＰ０～ＣＰ１１）である。また、４０はアドレスデコーダ（ＡＤＥＣ）であり、９ビットのデータサイズのアドレス情報が入力されると、１２本の出力のうちの１本だけに、ＤＦ０～ＤＦ１１のいずれかをアクティブにする出力が出される。さらに、３９はプライオリティ・エンコーダ（ＰＥ）であり、ＴＳＣ情報と組み合わされるアドレス情報ＴＭＰ０～ＴＭＰ３が出力される。

【００３５】このＩＤＴ１６において、端子３０にはＣＰＵ１０から出力された８ビットのデータサイズのＰＩＤ情報が、１３ビットのデータサイズに展開されて入力される。また、端子３２にはＣＰＵ１０から出力された９ビットのデータサイズのアドレス情報が入力され、ＡＤＥＣ４０においてデコードされる。このデコード出力はストローブ信号としてＤＦ０～ＤＦ１１に入力され、ＤＦ０～ＤＦ１１のうちのいずれか１つのみがラッチ可能とされる。この時、入力されたデータのＰＩＤ情報は、１２個のＤＦ０～ＤＦ１１に共通に入力されており、ＡＤＥＣ４０のデコード出力によりＤＦ０～ＤＦ１１が順次１つづつ選択されて、供給されているＰＩＤ情報がラッチされる。このようにして、ＤＦ０～ＤＦ１１に最大１２種類のＰＩＤ情報をラッチすることができ、ＤＦ０～ＤＦ１１によりＰＩＤテーブルが構成されるようになる。

【００３６】また、端子３３には１３ビットのデータサイズのＰＩＤ情報が入力され、比較器ＣＰ０～ＣＰ１１のＢ入力端子に共通に入力される。このＰＩＤ情報は、ＣＢＣモード解読化部Ｂにおいて、受信データであるパケットのヘッダから抽出されてＩＤＴ１６に供給された１５ビットのデータサイズのＰＩＤ／ＴＳＣ情報のうちの１３ビットのデータサイズのＰＩＤ情報である。この場合、比較器ＣＰ０～ＣＰ１１にはＤＦ０～ＤＦ１１よりのＰＩＤ情報がそれぞれＡ端子に入力されており、端子３３より入力されたＰＩＤ情報と一致するＰＩＤ情報がＡ端子に入力されている比較器から一致信号が出力される。従って、ＰＩＤテーブルにあるＰＩＤ情報が端子３３から入力されると、ＣＰ０～ＣＰ１１のいずれか１

つから一致信号が出力される。

【0037】このCP0～CP11の論理表を図4に示すが、CP0～CP11のA端子およびB端子にオール"1"が入力されていない時は、一致した時（A＝B）に"1"信号が出力される。また、一致してない時は"0"が出力される。さらに、PE39の論理表を図5に示す。このPE39の論理表において、例えば、CP2から"1"信号が出力された時は、PE39からは"0010"の4ビット（TMP0～TMP3）のデータが出力される。このPE39では図5に示すように、D0が最も優先された入力とされ、D11が最下位の優先度の入力とされる。ところで、DF0～DF11に設定されたPIDテーブルのPID情報と一致しないPID情報が入力されることがあるが、この場合にはPEにオール"0"が入力されることになる。この時、PEのNK出力が"1"となる。なお、NK出力が"1"の時には、後述するデータ鍵の鍵テーブルを読みに行くことが禁止される。

【0038】また、電源投入時等の初期処理時には端子31のレベルが"0"となり、DF0～DF11が全てプリセットされることから、DF0～DF11から"1"が出力される。従って、この場合はCP0～CP11からオール"0"が出力されることになり、上述のようにデータ鍵の鍵テーブルを読みに行くことが禁止される。なお、鍵テーブルを読みに行くことが禁止されている時は、鍵テーブルが使用されないので、この鍵テーブルにCPU10からデータ鍵を書き込むことができる。すなわち、データ鍵の初期設定をすることができる。また、端子31はパワーオンリセット端子であり、所定時間後にそのレベルは"1"に復帰する。

【0039】次に、受信データ中のPID情報とTSC情報により、データ鍵の鍵テーブルをサーチする動作を図6を参照しながら説明する。図6に示すステップS20にて、受信データのヘッダから13ビットのデータサイズのPID情報として"PID－F"が、2ビットのデータサイズのTSC情報として"10"が入力されたとする。この"10"は、データ鍵Ksでスクランブルされていることを示しているものとする。次いで、ステップS21にてこのPID情報はCP0～CP11のA端子に入力されて、DF0～DF11に格納されているPID情報と比較される。この結果、CP5から"1"信号が出力され、PE39のTEMP3～TEMP0の出力TMP［3．．0］が、"0101"となる。すなわち、"PID－F"がサーチされる。この、PE39の出力である"0101"は、アドレスHADDのの第4ビットから第1ビットとされる。

【0040】次いで、ステップS22にてTSC情報TSC［1．．0］の2ビット（"10"）がHADDの第5ビット，第6ビットとされて、6ビットサイズのアドレスHADD［8．．3］が生成される。したがって、アドレスHADD［8．．3］は"100101"となる。そして、ステップS23にて、生成された"100101"のアドレスHADD［8．．3］により鍵テーブルを参照すると、Ks_even Table から64ビット幅のデータ鍵Kse－Fが得られるようになる。そして、得られたデータ鍵Kse－Fに基づいて入力された暗号文の解読が解読装置Aで実行される。このように、本発明の解読化方法では間接的にデータ鍵を検索する間接検索方法としているので、PID情報に対応する13ビット幅のテーブルを用意することなく、6ビット幅の鍵テーブルを用意すればよいのでメモリの容量を削減して小型化することができる。

【0041】なお、このように解読中では鍵テーブルのKs_even Table，Ks_odd Table の一方しか使用されないため、使用していない鍵テーブルのデータ鍵を更新することができる。これは、前述したようにCPU10が実行するが、CPU10のタイミングと解読装置Aのタイミングとは非同期で動作するので、CPU10は非同期で更新するデータ鍵を書き込みに行く。このため、鍵テーブルの格納されるデュアルポートメモリ17においては、前述したように書き込みと読み出しが同時に同一アドレスで行われる場合が生じるのである。この時のタイミング例を図9に示すが、図示するタイミングで8クロック幅の反転REが発生した場合には、この8クロック期間においては、前述したようにCPU10からの同一アドレスの書き込みが禁止される。

【0042】ところで、何らかの原因によりPIDテーブルに同一のPIDが格納されている場合が生じる。例えば、1チャンネル当り最大12種類のPID情報とされるが、12種類のPIDを必要としない場合はPIDテーブルには12種類のPIDを書き込む必要はなく、必要する種類のPIDだけを書き込むことになる。すると、書き込まれていないPIDの欄のデータが偶然PIDと同じデータになることがある。このような場合には、誤ったデータ鍵を読み出して解読できなくなってしまう恐れがあるので、本発明においては次のようにしてこれを防止している。

【0043】図7に示すようにPIDテーブルに複数の"PID－F"がある場合は、その内のアドレスHADDの小さい方を優先するようにしている。これは、アドレスHADDの小さい方からPIDテーブルに書き込みに行くので、アドレスHADDの大きい方が誤っている確率が高いからである。これにより、PID情報として"PID－F"が入力されると、アドレスHADD［8．．1］として"01000101"が得られるようになり、前述した処理と同様の処理が行われ、Ks_even Table から64ビット幅のデータ鍵Kse－Fが得られるようになる。

【0044】また、入力データのヘッダからのPID情報により、PIDテーブルを参照しても該当するPID

がない場合がある。例えば、図８に示すようにＰＩＤ情報として”ＰＩＤ－Ｆ”が入力されても、ＰＩＤテーブル中には”ＰＩＤ－Ｆ”に該当するＰＩＤがない。このような場合には、鍵テーブルのKs_even Table，Ks_odd Table のいずれも読みに行くことなく、データ鍵を読み出さない。この場合は、入力データであるＴＳはそのまま出力することになる。

【００４５】次に、図１に示す解読装置Ａの解読フローを図１０に示すが、トランスポートストリーム（ＴＳ）が入力されると、ステップＳ１０にてスクランブルされているか否かが判定される。この判定はスクランブルされたことを示すヘッダ中のフラグが立っているか否かを検出することにより判定される。この場合、フラグが立っている場合はスクランブルｏｎと判定されて、ステップＳ１１に進み、ここで所望のフラグ等の書き換えが行われる。次いで、ステップＳ１２にてヘッダからＰＩＤ情報が抽出されて、鍵テーブルが参照される。この場合の鍵テーブルは、鍵テーブル処理のステップＳ１６にてホストインターフェース処理が行われて、ステップＳ１７にて書き込まれた鍵テーブルが参照される。なお、ステップＳ１２の処理は、前記図６に示す処理である。

【００４６】以上のステップＳ１０ないしステップＳ１２の処理がヘッダコントロール処理である。なお、ステップＳ１０にてスクランブルｏｆｆと判定された場合には、ＴＳはそのまま出力される。次いで、ステップＳ１３にてＣＢＣモードの解読処理が行われ、ステップＳ１４にて解読処理される暗号文が６４ビットの整数倍か否かが判定される。暗号文に端数がありＮｏと判定された場合は、ステップＳ１５にて端数部分についてＯＦＢモードの解読処理が行われ、解読処理された平文が出力される。また、ステップＳ１４にて６４ビットの整数倍と判定された場合は、解読処理された６４ビットの平文が出力されるようになる。

【００４７】ところで、図１０に示すような解読フローの実行は、前記図１８に示す解読化アルゴリズムにより実行されている。図１８に示す解読化アルゴリズムは前述したとおりであるのでここでは省略するが、解読化アルゴリズム中の解読アルゴリズムおよび暗号アルゴリズムを実行する構成の詳細を図１１ないし図１５を参照して説明する。図１１は暗号アルゴリズムを実行する暗号処理の構成を示す。図１１において、６４ビット幅の入力データは、上位３２ビットのデータと下位の３２ビットのデータに分割されて最初の暗号８段に入力される。この暗号８段は、関数の演算を行う４段の演算段が２回繰り返された構成とされる。そして、入力された上位３２ビットのデータと下位の３２ビットのデータに、演算段２０の初段において関数π１の演算が施される。ついで、第２段において初段の出力に関数π２の演算が施される。この場合、第２段には３２ビット幅のワーク鍵Ｋ１が入力され、このワーク鍵Ｋ１を用いて第２段の演算が行われている。

【００４８】さらに、第３段において第２段の出力に関数π３の演算が施される。この場合、第３段には３２ビット幅のワーク鍵Ｋ２，Ｋ３が入力され、このワーク鍵Ｋ２，Ｋ３を用いて演算が行われている。続いて、第４段において第３段の出力に関数π４の演算が施される。この場合、第４段には３２ビット幅のワーク鍵Ｋ４が入力され、このワーク鍵Ｋ４を用いて演算が行われている。さらに残る４段の演算を行う演算段２１において、演算段２０からの出力に初段において関数π１の演算が施される。ついで、第２段において初段の出力に関数π２の演算が施される。この場合、第２段には３２ビット幅のワーク鍵Ｋ５が入力され、このワーク鍵Ｋ５を用いて演算が行われている。

【００４９】さらに、第３段において第２段の出力に関数π３の演算が施される。この場合、第３段には３２ビット幅のワーク鍵Ｋ６，Ｋ７が入力され、このワーク鍵Ｋ６，Ｋ７を用いて演算が行われている。続いて、第４段において第３段の出力に関数π４の演算が施される。この場合、第４段には３２ビット幅のワーク鍵Ｋ８が入力され、このワーク鍵Ｋ８を用いて演算が行われている。このようにして暗号処理の行われた上位３２ビット、下位３２ビットの合計６４ビット幅のデータは、さらに暗号８段２２に入力される。この暗号８段２２において、上述した暗号８段の演算と同様の演算が施されて、上位３２ビット、下位３２ビットの合計６４ビット幅のランダム化された出力データが得られる。

【００５０】なお図示しているように、暗号８段の繰返し数は２回に限らず、所望の回数繰り返すことができる。この回数を多く繰り返すほど、出力データは高度にランダム化されて、暗号強度を強いものとすることができる。なお、演算段の格段で行われている関数の演算は、一定の規則に従ってある文字を他の文字に置き換える換字と、文字の順序を入れ替える転置を行う演算とされている。

【００５１】次に、解読アルゴリズムを実行する解読処理の構成を図１２に示すが、前述した暗号処理と異なる点は、暗号処理の構成の出力側から逆に演算を行うようにしている点である。すなわち、暗号８段のうちの最初の４段の演算段２３においては、上位３２ビットと下位３２ビットに分割された６４ビット幅の暗号化されている入力データに、３２ビット幅のワーク鍵Ｋ８を用いて関数π４の演算を施している。次いで、第２段目において、初段の出力データにワーク鍵Ｋ７を用いて関数π３の演算を施している。さらに、第３段目において、第２段の出力データにワーク鍵Ｋ６，Ｋ７を用いて関数π２の演算を施している。さらにまた第３段目において、第２段の出力データにワーク鍵Ｋ５を用いて関数π２の演算を施し、第４段目において、第３段の出力データに関数π１の演算を施している。

【００５２】このような４段の演算が、演算２４においてワーク鍵Ｋ４～Ｋ１を用いて同様に行われる。さらに、上記した暗号８段の演算が縦続されている暗号８段２５においても実行されて、解読化された上位３２ビット、下位３２ビットの計６４ビット幅の出力データが得られるようになる。なお、暗号８段の繰り返し回数は、暗号処理において実行された暗号８段の繰返し回数と同じ回数とされる。

【００５３】次に、演算段で行われている演算の詳細を暗号処理の演算段２０を例に上げて図１３を参照しながらに詳細に説明する。初段の関数π１の演算では、入力された３２ビットに分割された上位ビットは、演算されることなくそのまま出力され、上位ビットと下位ビットの排他的論理和が演算されて下位ビットとして出力される。続く、第２段の関数π２の演算では、下位３２ビットのデータｘにワーク鍵Ｋ１が加算されて、ｘ＋Ｋ１がまず演算される。次いで、ｘ＋Ｋ１をｙとした時に、ｙを１ビット左巡回シフトし、その値にｙ－１を加算してｚを得る。次に、ｚを４ビット左巡回シフトし、その値とｚとの排他的論理和を得る。この演算結果と、上位３２ビットの排他的論理和が演算されて、演算された上位３２ビットのデータが出力される。この場合、下位３２ビットは入力されたデータが、演算されることなくそのまま出力される。

【００５４】また、第３段の関数π３の演算では、上位３２ビットのデータｘにワーク鍵Ｋ２が加算されて、ｘ＋Ｋ２がまず演算される。次いで、ｘ＋Ｋ２をｙとした時に、ｙを２ビット左巡回シフトし、その値にｙ＋１を加算してｚを得る。次に、ｚを８ビット左巡回シフトし、その値とｚとの排他的論理和ａを得る。さらに、ａにワーク鍵Ｋ３が加算されて、ａ＋Ｋ３が演算される。次いで、ａ＋Ｋ３をｂとした時に、ｂを１ビット左巡回シフトし、その値にｂを加算してｃを得る。次に、ａとｘのビット毎の論理和と、ｃを１６ビット左巡回シフトした値との排他的論理和を演算する。この演算結果と、下位３２ビットのデータとの排他的論理和を演算して、演算された下位３２ビットのデータを出力する。なお、上位３２ビットのデータは、演算されることなくそのまま上位３２ビットの出力データとなる。

【００５５】さらにまた、第４段の関数π４の演算では、下位３２ビットのデータｘにワーク鍵Ｋ４が加算されて、ｘ＋Ｋ４がまず演算される。次いで、ｘ＋Ｋ４をｙとした時に、ｙを２ビット左巡回シフトし、その値にｙ＋１を加算する。この演算結果と、上位３２ビットの排他的論理和が演算されて、演算された上位３２ビットのデータが出力される。この場合、下位３２ビットのデータは演算されることなく、そのまま下位３２ビットのデータとして出力される。

【００５６】上記演算において、ワーク鍵Ｋ１～Ｋ４をデータに加算することにより、文字を他の文字で置き換える換字処理が行われ、データを巡回シフトさせることにより文字の位置を入れ替える転置が行われる。このように、換字と転置のアルゴリズムを行うことにより平文が暗号文に暗号化される。また、解読化する場合には、暗号化と逆の換字と転置のアルゴリズムを行うことにより元の平文に解読することができる。

【００５７】次に、上述した関数の演算を行う構成をさらに詳細に説明するが、関数π２の例を図１４に上げて説明するものとする。図１４において、第１加算器Ａｄｄ８０において、下位３２ビットの入力データｘと３２ビットのワーク鍵Ｋ１とが加算され、加算データｙが出力される。この加算データｙは第１左巡回シフター８１において１ビット左巡回シフトされると共に、第２加算器８２において、第１左巡回シフター８１の出力と加算される。この加算結果に第３加算器８４においてー１が加算されて、データｚが演算される。このデータｚは第２左巡回シフター８５において４ビット左巡回シフトされると共に、排他的論理和回路８６に供給される。この排他的論理和回路８６には第２左巡回シフター８５の出力データ、データｚ、上位３２ビット入力データが入力され、３つのデータの排他的論理和が演算される。この演算結果は、次段に入力される上位３２ビット入力データとなる。また、下位３２ビット入力データは、演算されることなく次段に入力される下位３２ビット入力データとなる。

【００５８】次に、６４ビット幅のデータ鍵と２５６ビット幅のシステム鍵から２５６ビット幅のワーク鍵を生成する鍵スケジュール処理の構成を図１５に示す。鍵スケジュール処理は図１５に示すように４段の演算段２６，２７が２段と、１段の演算段２８が１段縦続接続された構成とされている。また、４段の演算段２６，２７においては、初段において関数π１の演算が行われ、２段目において関数π２の演算が行われ、３段目において関数π３の演算が行われ、４段目において関数π４の演算が行われている。

【００５９】このような演算アルゴリズムは、前述した暗号処理のアルゴリズムと同じであるのでその説明は省略するが、鍵スケジュール処理においては、入力データが６４ビットのデータ鍵とされ、それぞれ３２ビットのシステム鍵Ｊ１～Ｊ８を用いて関数π１ないし関数π４の演算が行われて、それぞれ３２ビットの８つのワーク鍵Ｋ１～Ｋ８が生成されている。ただし、全体で９段の演算を行っており、最終段において関数π１の演算を行う点で、前述した暗号処理のアルゴリズムと相違している。

【００６０】なお、演算段２６の関数π２演算後の上位３２ビット出力データがワーク鍵Ｋ１として出力され、関数π３演算後の下位３２ビット出力データがワーク鍵Ｋ２として出力され、関数π４演算後の上位３２ビット出力データがワーク鍵Ｋ３として出力されている。さら

に、演算段２７の関数π１演算後の下位３２ビット出力データがワーク鍵Ｋ４として出力され、関数π２演算後の上位３２ビット出力データがワーク鍵Ｋ５として出力され、関数π３演算後の上位３２ビット出力データがワーク鍵Ｋ６として出力され、関数π４演算後の上位３２ビット出力データがワーク鍵Ｋ７として出力され、最終段２８の関数π１演算後の下位３２ビット出力データがワーク鍵Ｋ８として出力されている。

【００６１】上述した、図１１に示す暗号処理と図１３に示す鍵スケジュール処理を参照すると、４段の演算段の構成、すなわち演算アルゴリズムは等しくされており、この４段の演算段の演算を繰り返し行うことにより、暗号処理あるいは鍵スケジュール処理を実行することができる。このことから、演算コアを図１６（ａ）に示すように、関数π１の演算段、関数π２の演算段、関数π３の演算段、関数π４の演算段を縦続接続した構成とすれば、演算コアを繰返し実行することで、暗号処理あるいは鍵スケジュール処理を実行することができる。なお、この演算コアは、図２０に示すデータ鍵とシステム鍵からワーク鍵を生成すると共に、ＣＢＣモードおよびＯＦＢモードで暗号処理を行っているEncryptor に相当し、Encryptor コアとされる。この場合、Encryptor コアにはデータ鍵Ｋｓ１～Ｋｓ４とデータ鍵Ｋｓ５～Ｋｓ８とが時分割で供給される。

【００６２】また、図１２を参照すると、４段の演算段の構成、すなわち演算アルゴリズムは等しくされており、この４段の演算段の演算を繰り返し行うことにより、解読処理を実行することができる。このことから、解読演算コアを図１６（ｂ）に示すように、関数π４の演算段、関数π３の演算段、関数π２の演算段、関数π１の演算段を縦続接続した構成とすれば、解読演算コアを繰返し実行することで、解読処理を実行することができる。なお、この解読演算コアは、図２０に示すＣＢＣモードおよびＯＦＢモードの解読処理を行っているDecryptor に相当し、Decryptor コアとされる。この場合、Decryptor コアにはデータ鍵Ｋｓ８～Ｋｓ５とデータ鍵Ｋｓ４～Ｋｓ１とが時分割で供給される。このように、Encryptor コアを繰返し実行することにより暗号処理および鍵スケジュールのアルゴリズムを実行することができ、Decryptor コアを繰返し実行することにより解読アルゴリズムを実行することができる。

【００６３】以上、本発明の解読化方法を実行する解読装置の説明をしたが、本発明の電子機器は、このような解読装置を少なくとも備えているチューナーやテレビジョン装置等である。また、以上の説明では６４ビットブロックの暗号文を６４ビットのデータ鍵、および２５６ビットのシステム鍵を用いて６４ビットブロックの平文を生成するものとして説明したが、本発明がこれらの数値に限定されるものではなく、任意の数値とすることができる。さらに、本発明は上述した転置および換字を繰り返すような暗号化・解読化方式に限定されるものではなく、他の暗号化・解読化方式にも適用することができる。

【００６４】

【発明の効果】本発明は以上のように構成されているので、解読処理を行うための鍵情報が格納されているメモリ手段の読み出しおよび書き込み制御を適切に行うことができ、メモリ制御を容易に行うことのできる解読化方法とすることができる。したがって、受信側においてリアルタイムで全ての解読処理を実行することのできる解読化方法とすることができる。また、このような解読化方法を実行する解読手段を備える電子機器におけるメモリ制御手段の構成を小さくすることができるので、電子機器を小型かつ安価に提供することができるようになる。

【図面の簡単な説明】

【図１】本発明の解読化方法の実施の形態である解読装置の構成例を示すブロック図である。

【図２】図１に示す解読装置におけるＩＤＴの構成を示すブロック図である。

【図３】図１に示すＣＰＵから見た解読装置におけるレジスタのメモリ空間を示す図表である。

【図４】図２に示すＩＤＴの比較器ＣＰ０～ＣＰ１１の論理表を示す図表である。

【図５】図２に示すＩＤＴのＰＥの論理表を示す図表である。

【図６】図１に示す解読装置において、入力データのヘッダ中の情報から間接検索方法によりデータ鍵をサーチする方法を説明するための図である。

【図７】ＰＩＤテーブル中にＰＩＤが重複している場合の動作を説明するための図である。

【図８】ＰＩＤテーブル中に該当するＰＩＤが存在しない場合の動作を説明するための図である。

【図９】図１に示す解読装置におけるＤＰＭＥＭのリードタイミングの例を示す図である。

【図１０】図１に示す解読装置の解読フローを示すフローチャートである。

【図１１】暗号処理の構成を示す図である。

【図１２】解読処理の構成を示す図である。

【図１３】暗号処理における基本関数の詳細を示す図である。

【図１４】基本関数中の関数π２を演算するための詳細な構成を示す図である。

【図１５】鍵スケジュール処理の構成を示す図である。

【図１６】Encryptor コアとDecryptor コアの構成を示す図である。

【図１７】従来の暗号化のアルゴリズムを示す図である。

【図１８】従来の解読化のアルゴリズムを示す図である。

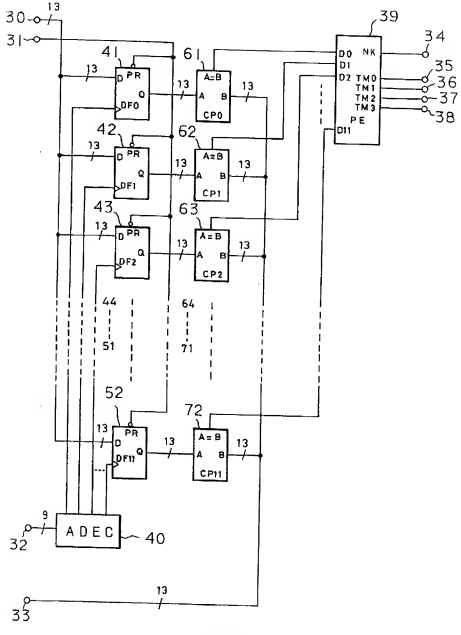
【図１９】ＣＢＣモードとＯＦＢモードの暗号化利用モードの構成を示す図である。

【図２０】従来の暗号化・解読化方式の構成を示す図である。

【符号の説明】

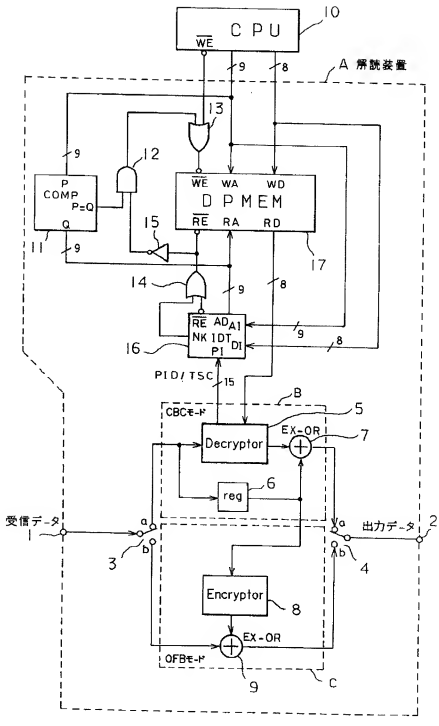
１　受信データ、２　出力データ、３，４　切り換え手段、５　Decryptor 、６レジスタ、７，９，８６　排他的論理和回路、８　Encryptor 、１０　ＣＰＵ、１１　比較器、１２　アンドゲート、１３，１４　オアゲート、１５　インバータ、１６　ＩＤＴ、１７　ＤＰＭＥＭ、２０～２８　演算段、３９　ＰＥ、４０　ＡＤＥＣ、４１～５２　フリップフロップ、６１～７２　比較器、８０，８２，８４　加算器、８１，８５　左巡回シフター

【図２】



IDTブロック図

【図1】


CPU
10
WE
A 解読装置
9
8
13
12
P
COMP
P=Q
Q
11
WE WA WD
D P MEM
RE RA RD
15
17
14
RE AD
NK IDT
PI DI
16
PID/TSC
15
CBCモード
5
Decryptor
EX-OR
7
reg
6
受信データ
1
出力データ
2
a
b
3
4
Encryptor
8
EX-OR
OFBモード
9
C

【図3】

HADD[8..0]	内容	ビット数
0 0010 0×××	CBC Initial value table	64bit× 1
0 010× ××××	SYSTEM_Key table	256bit× 1
0 100× ××××	PID value table	13bit×12
1 0××× ××××	Ks_even value table	64bit×12
1 1××× ××××	Ks_odd value table	64bit×12

【図4】

CP0～CP11の論理表

入力 A, B	出力 A=B
A=B(AもBもall'1'でないとき)	1
A=B(AもBもall'1'のとき)	0
A<B	0
A>B	0

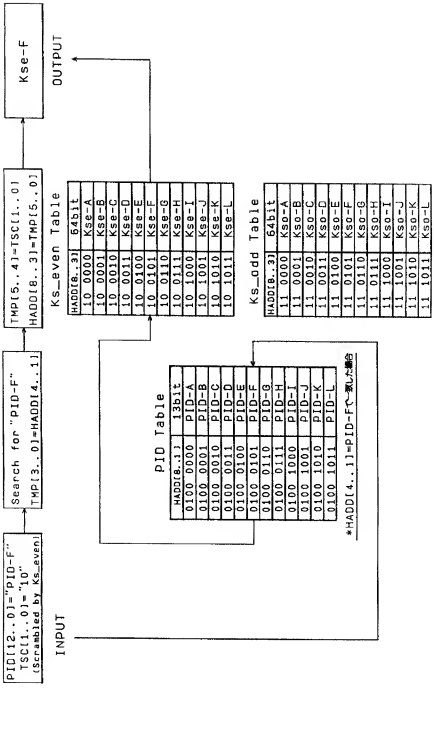
【図5】

PEの論理表

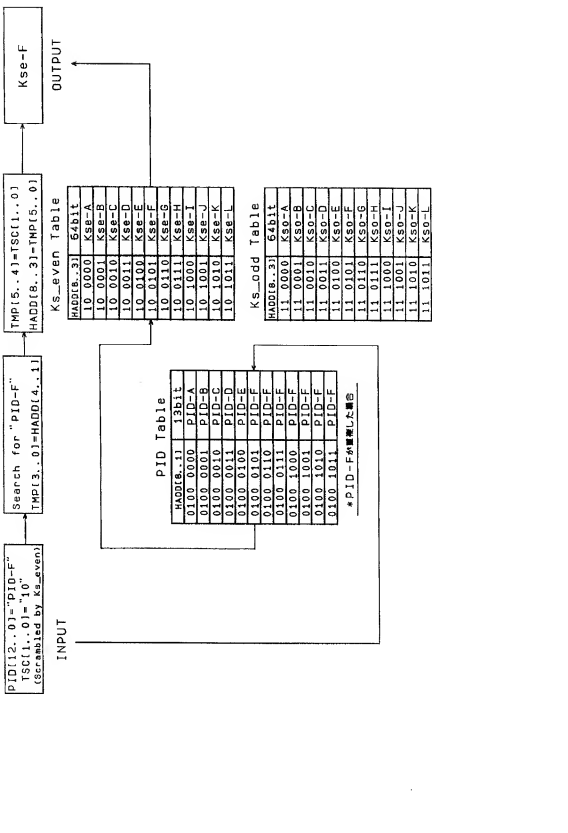
入力												出力				
D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0	TMP3	TMP2	TMP1	TMP0	NK
×	×	×	×	×	×	×	×	×	×	×	1	0	0	0	0	0
×	×	×	×	×	×	×	×	×	×	1	0	0	0	0	1	0
×	×	×	×	×	×	×	×	×	1	0	0	0	0	1	0	0
×	×	×	×	×	×	×	×	1	0	0	0	0	0	1	1	0
×	×	×	×	×	×	×	1	0	0	0	0	0	1	0	0	0
×	×	×	×	×	×	1	0	0	0	0	0	0	1	0	1	0
×	×	×	×	×	1	0	0	0	0	0	0	0	1	1	0	0
×	×	×	×	1	0	0	0	0	0	0	0	0	1	1	1	0
×	×	×	1	0	0	0	0	0	0	0	0	1	0	0	0	0
×	×	1	0	0	0	0	0	0	0	0	0	1	0	0	1	0
×	1	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0
1	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	0
0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1

×=Do not care

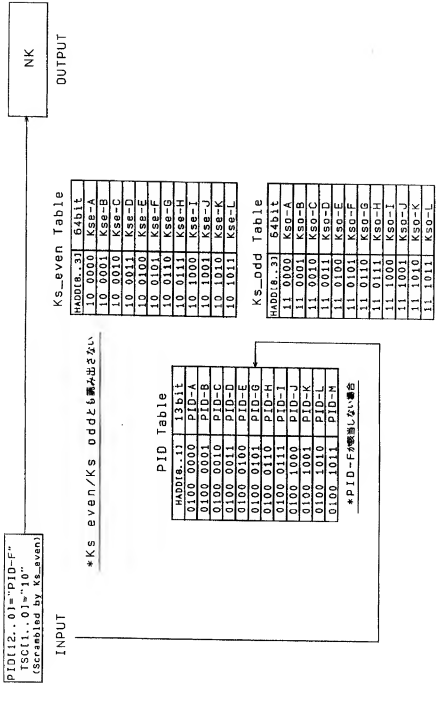
【図6】


INPUT
S20
PID[12..0]="PID-F"
TSC[1..0]="10"
(Scrambled by Ks_even)
S21
Search for "PID-F"
TMP[3..0]=HADDI[4..1]
PID Table
HADDI[8..1] 13bit
0100 0000 PID-A
0100 0001 PID-B
0100 0010 PID-C
0100 0011 PID-D
0100 0100 PID-E
0100 0101 PID-F
0100 0110 PID-G
0100 0111 PID-H
0100 1000 PID-I
0100 1001 PID-J
0100 1010 PID-K
0100 1011 PID-L
*HADDI[4..1]=PID-Fに一致した番号
TMP[5..4]=TSC[1..0]
HADDI[8..3]=TMP[5..0]
S22
Ks_even Table
HADDI[8..3] 64bit
10 0000 Kse-A
10 0001 Kse-B
10 0010 Kse-C
10 0011 Kse-D
10 0100 Kse-E
10 0101 Kse-F
10 0110 Kse-G
10 0111 Kse-H
10 1000 Kse-I
10 1001 Kse-J
10 1010 Kse-K
10 1011 Kse-L
Ks_odd Table
HADDI[8..3] 64bit
11 0000 Kso-A
11 0001 Kso-B
11 0010 Kso-C
11 0011 Kso-D
11 0100 Kso-E
11 0101 Kso-F
11 0110 Kso-G
11 0111 Kso-H
11 1000 Kso-I
11 1001 Kso-J
11 1010 Kso-K
11 1011 Kso-L
S23
Kse-F
OUTPUT

【図7】


INPUT
S20
PID[12..0]="PID-F"
TSC[1..0]="10"
(Scrambled by Ks_even)
S21
Search for "PID-F"
TMPI3..0]=HADDI4..1]
PID Table
HADDI8..1] 13bit
0100 0000 PID-A
0100 0001 PID-B
0100 0010 PID-C
0100 0011 PID-D
0100 0100 PID-E
0100 0101 PID-F
0100 0110 PID-F
0100 0111 PID-F
0100 1000 PID-F
0100 1001 PID-F
0100 1010 PID-F
0100 1011 PID-F
*PID-Fが検出した場合
S22
TMPI5..4]=TSC[1..0]
HADDI6..3]=TMPI5..0]
Ks_even Table
HADDI6..3] 64bit
10 0000 Kse-A
10 0001 Kse-B
10 0010 Kse-C
10 0011 Kse-D
10 0100 Kse-E
10 0101 Kse-F
10 0110 Kse-G
10 0111 Kse-H
10 1000 Kse-I
10 1001 Kse-J
10 1010 Kse-K
10 1011 Kse-L
Ks_odd Table
HADDI6..3] 64bit
11 0000 Kso-A
11 0001 Kso-B
11 0010 Kso-C
11 0011 Kso-D
11 0100 Kso-E
11 0101 Kso-F
11 0110 Kso-G
11 0111 Kso-H
11 1000 Kso-I
11 1001 Kso-J
11 1010 Kso-K
11 1011 Kso-L
S23
Kse-F
OUTPUT

【図8】


INPUT
PID[12..0]="PID-F"
TSC[1..0]="10"
(Scrambled by Ks_even)
*Ks even/Ks odd も異なる
PID Table
HADDR[8..3] | 13bit
0100 0000 | PID-A
0100 0001 | PID-B
0100 0010 | PID-C
0100 0011 | PID-D
0100 0100 | PID-E
0100 0101 | PID-F
0100 0110 | PID-G
0100 0111 | PID-I
0100 1000 | PID-J
0100 1001 | PID-K
0100 1010 | PID-L
0100 1011 | PID-M
*PID-Fが異なる場合
Ks_even Table
HADDR[8..3] | 64bit
10 0000 | Ks0-A
10 0001 | Ks0-B
10 0010 | Ks0-C
10 0011 | Ks0-D
10 0100 | Ks0-E
10 0101 | Ks0-F
10 0110 | Ks0-G
10 0111 | Ks0-H
10 1000 | Ks0-I
10 1001 | Ks0-J
10 1010 | Ks0-K
10 1011 | Ks0-L
Ks_odd Table
HADDR[8..3] | 64bit
11 0000 | Ks0-A
11 0001 | Ks0-B
11 0010 | Ks0-C
11 0011 | Ks0-D
11 0100 | Ks0-E
11 0101 | Ks0-F
11 0110 | Ks0-G
11 0111 | Ks0-H
11 1000 | Ks0-I
11 1001 | Ks0-J
11 1010 | Ks0-K
11 1011 | Ks0-L
NK
OUTPUT

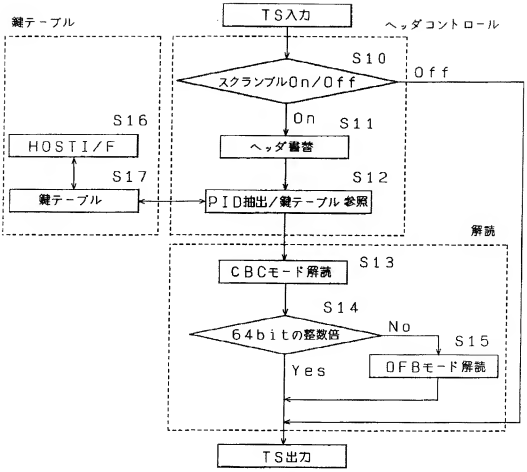
【図9】

8クロック
(a) 同期信号
(b) クロック
(c) アドレス (9bit)
(d) データ (8bit)
(e) RE
203 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 194 195 196 197 198 199 200 201

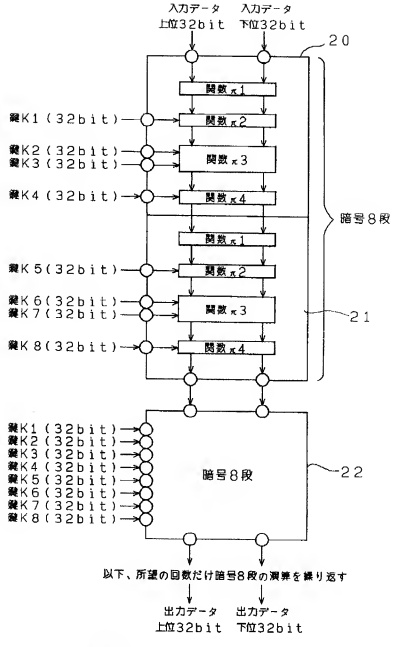
【図17】

システム鍵 (256bit)
A
J
データ鍵 (64bit)
B
Ks
暗号アルゴリズム
C
D
ワーク鍵 (256bit)
Kw
平文 (64bit)
E
F
暗号アルゴリズム
OFBモード
G
暗号アルゴリズム
Kw
CBCモード
暗号文 (64bit)
H
暗号化

【図10】

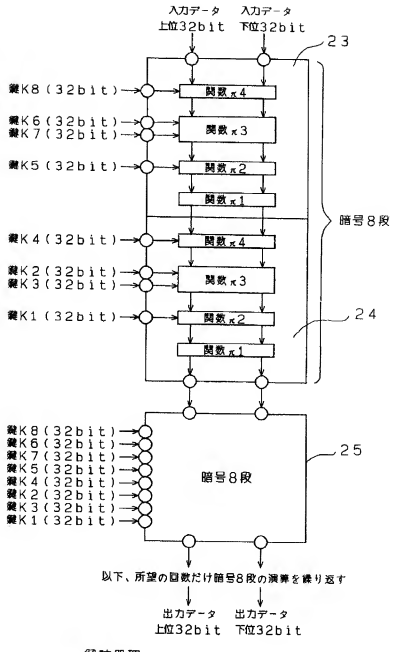

鍵テーブル
TS入力
ヘッダコントロール
S10
スクランブルOn/Off
Off
On
S11
ヘッダ書替
S16
HOSTI/F
S17
鍵テーブル
S12
PID抽出/鍵テーブル参照
解読
S13
CBCモード解読
S14
64bitの整数倍
No
S15
OFBモード解読
Yes
TS出力

【図11】


入力データ
上位32bit
入力データ
下位32bit
20
関数πi
鍵K1（32bit）
関数π2
鍵K2（32bit）
鍵K3（32bit）
関数π3
鍵K4（32bit）
関数π4
暗号8段
関数πi
鍵K5（32bit）
関数π2
鍵K6（32bit）
鍵K7（32bit）
関数π3
21
鍵K8（32bit）
関数π4
鍵K1（32bit）
鍵K2（32bit）
鍵K3（32bit）
鍵K4（32bit）
鍵K5（32bit）
鍵K6（32bit）
鍵K7（32bit）
鍵K8（32bit）
暗号8段
22
以下、所望の回数だけ暗号8段の演算を繰り返す
出力データ
上位32bit
出力データ
下位32bit

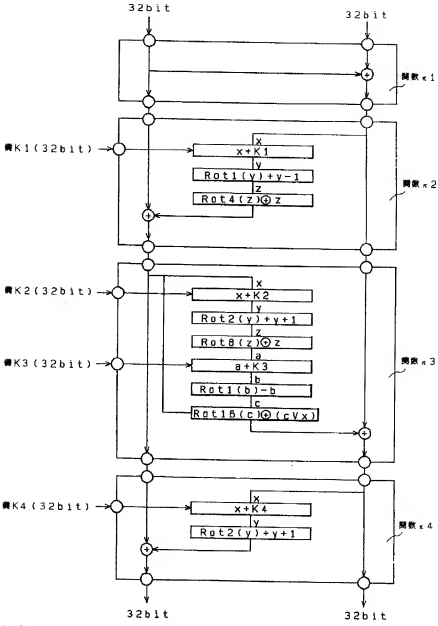
暗号処理

【図12】


入力データ
上位32bit
入力データ
下位32bit
23
鍵K8（32bit）
関数×4
鍵K6（32bit）
鍵K7（32bit）
関数×3
鍵K5（32bit）
関数×2
関数×1
暗号8段
鍵K4（32bit）
関数×4
鍵K2（32bit）
鍵K3（32bit）
関数×3
鍵K1（32bit）
関数×2
24
関数×1
鍵K8（32bit）
鍵K6（32bit）
鍵K7（32bit）
鍵K5（32bit）
鍵K4（32bit）
鍵K2（32bit）
鍵K3（32bit）
鍵K1（32bit）
暗号8段
25
以下、所望の回数だけ暗号8段の演算を繰り返す
出力データ
上位32bit
出力データ
下位32bit

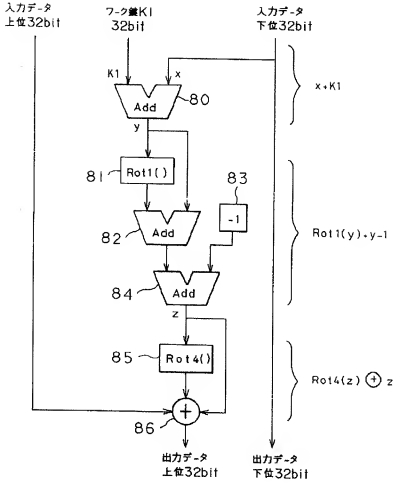
解読処理

【図13】


32bit
32bit
関数ｋ1
鍵K1(32bit)
x
x+K1
y
Rot1(y)+y-1
z
Rot4(z)⊕z
関数ｋ2
鍵K2(32bit)
x
x+K2
y
Rot2(y)+y+1
z
Rot8(z)⊕z
鍵K3(32bit)
a
a+K3
b
Rot1(b)-b
c
Rot15(c)⊕(cVx)
関数ｋ3
鍵K4(32bit)
x
x+K4
y
Rot2(y)+y+1
関数ｋ4
32bit
32bit
〔記号〕
⊕　：ビット毎の排他的論理和　　＋：2^{32}を法とした加算　　－：2^{32}を法とした減算
Rotn( )：左巡回nビットシフト　　V：ビット毎の論理和

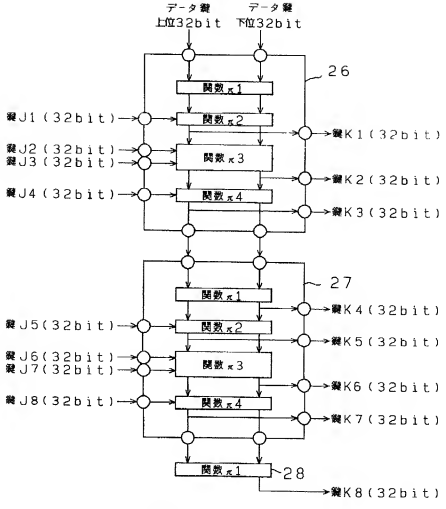
基本関数

【図14】



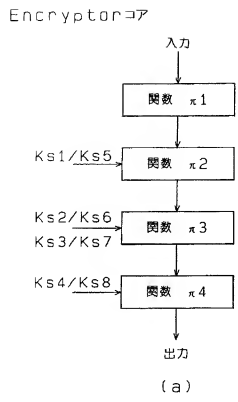
基本関数π2の回路構成例

【図15】

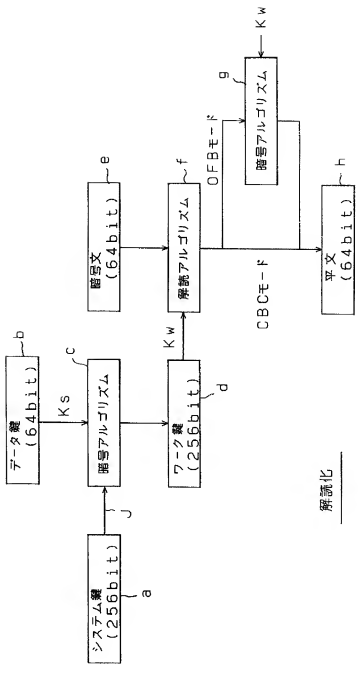


鍵スケジュール処理

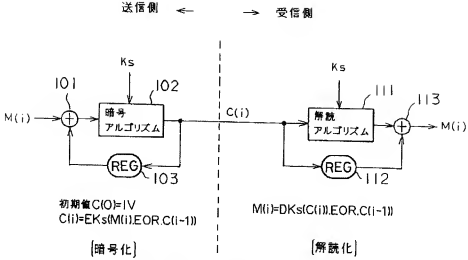
【図16】


Encryptorコア
入力
関数 π1
Ks1/Ks5
関数 π2
Ks2/Ks6
Ks3/Ks7
関数 π3
Ks4/Ks8
関数 π4
出力
(a)
Decryptorコア
入力
Ks8/Ks4
関数 π4
Ks6/Ks2
Ks7/Ks3
関数 π3
Ks5/Ks1
関数 π2
関数 π1
出力
(b)

【図18】


システム鍵
(256bit)
a
J
データ鍵
(64bit)
b
Ks
暗号アルゴリズム
c
ワーク鍵
(256bit)
d
Kw
暗号文
(64bit)
e
解読アルゴリズム
f
OFBモード
暗号アルゴリズム
g
Kw
CBCモード
平文
(64bit)
h
解読化

【図19】

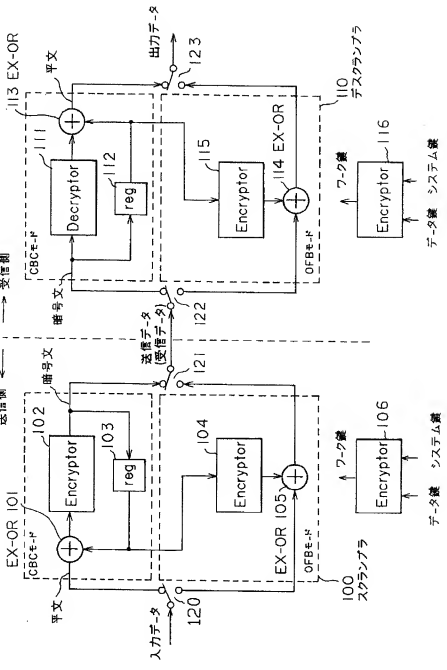


[記号]

Ks　：データ鍵
M(i)　：i番目の平文ブロック
C(i)　：i番目の暗号文ブロック
IV　：スクランブル用の初期値
Ek(m)　：kでmを暗号化
Dk(c)　：kでcを解読化

暗号化利用モード

【図２０】


送信側
受信側
EX-OR 101
CBCモード
Encryptor
102
reg
103
104
EX-OR 105
OFBモード
Encryptor
106
ワーク鍵
データ鍵
システム鍵
100
スクランブラ
平文
暗号文
入力データ
120
送信データ
(受信データ)
121
122
123
出力データ
113 EX-OR
Decryptor
111
112
115
114 EX-OR
110
デスクランブラ
116